Non-Reversible Parallel Tempering: a Scalable Highly Parallel MCMC Scheme

Saifuddin Syed[*], Alexandre Bouchard-Côté[*], George Deligiannidis[†], Arnaud Doucet[†]

October 8, 2019

Abstract

Parallel tempering (PT) methods are a popular class of Markov chain Monte Carlo schemes used to sample complex high-dimensional probability distributions. They rely on a collection of N interacting auxiliary chains targeting tempered versions of the target distribution to improve the exploration of the state-space. We provide here a new perspective on these highly parallel algorithms and their tuning by identifying and formalizing a sharp divide in the behaviour and performance of reversible versus non-reversible PT schemes. By analyzing the behaviour of PT algorithms using a novel asymptotic regime in which N goes to infinity, we show indeed that a class of non-reversible PT methods dominates its reversible counterparts and identify distinct scaling limits for the non-reversible and reversible schemes, the former being a piecewise-deterministic Markov process and the latter a diffusion. In particular, a major limitation of reversible PT is that its performances eventually collapse as N increases whereas those of non-reversible PT improve. These theoretical results are exploited to develop an adaptive non-reversible PT scheme approximating the optimal annealing schedule. We provide a wide range of numerical examples supporting our theoretical and methodological contributions.

1 Introduction

Markov Chain Monte Carlo (MCMC) methods are widely used to approximate expectations with respect to a probability distribution with density $\pi(x)$ known up to a normalizing constant, i.e., $\pi(x) = \gamma(x)/\mathcal{Z}$ where γ can be evaluated pointwise but the normalizing constant $\mathcal{Z}$ is unknown. When π has multiple well-separated modes, highly varying curvature or when one is interested in sampling over combinatorial spaces, standard MCMC algorithms can perform very poorly. This work is motivated by the need for practical methods for these difficult sampling problems. A natural direction to address them is to use multiple cores and to distribute the computation.

[*]Department of Statistics, University of British Columbia, Canada.
[†]Department of Statistics, University of Oxford, UK.

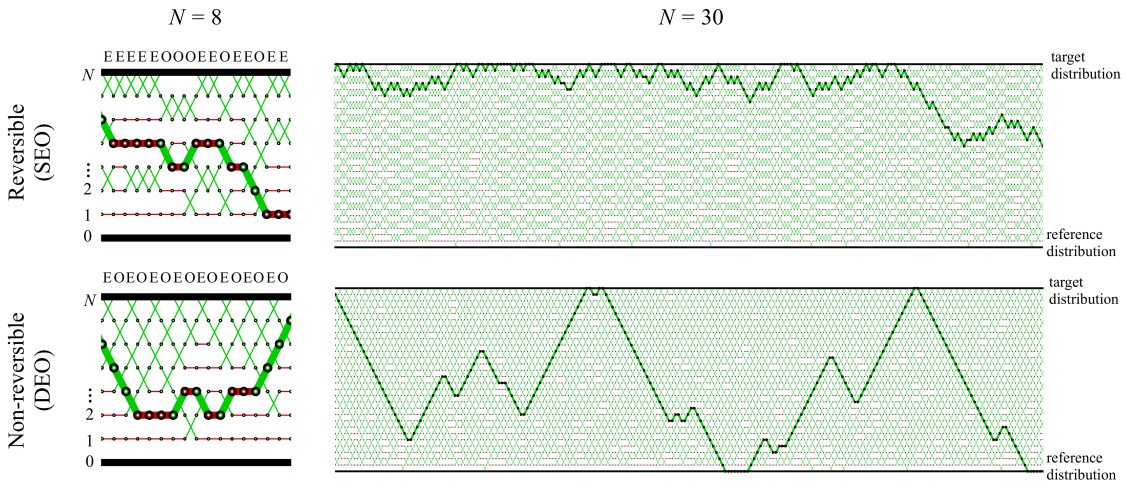

Figure 1: Reversible (top) and non-reversible (bottom) PT for $N = 8$ (left) and $N = 30$ auxiliary chains (right) using equally spaced annealing parameters on a Bayesian change-point detection model [Davidson-Pilon, 2015] where π_0 is the prior, π the posterior. The sequence of moves forms $N + 1$ index trajectories (paths formed by the red and green edges). We show one such paths in bold. The annealing schedule is clearly suboptimal as most swaps between the $\beta = 0$ and $\beta = 1/N$ chains are rejected. This is corrected by adaptive tuning (Section 5.4).

1.1 Parallel Tempering

One popular approach for multi-core and distributed exploration of complex distributions is Parallel Tempering (PT) which was introduced independently in statistics [Geyer, 1991] and physics [Hukushima and Nemoto, 1996]; see also [Swendsen and Wang, 1986] for an earlier related proposal. Since its inception, PT remains to this day the go-to "workhorse" MCMC method to sample from complex multi-modal target distributions arising in physics, chemistry, biology, statistics, and machine learning; see, e.g., [Desjardins et al., 2014, Cho et al., 2010, Earl and Deem, 2005, Andrec et al., 2005, Pitera and Swope, 2003, Cheon and Liang, 2008]. A recent empirical benchmark shows PT methods consistently outperform other state-of-the-art sampling methods [Ballnus et al., 2017].

To sample from the target distribution π, PT introduces a sequence of auxiliary *tempered* or *annealed* probability distributions with densities $\pi^{(\beta_i)}(x) \propto L(x)^{\beta_i}\pi_0(x)$ for $i = 0, 1, ..., N$, where π_0 is an easy-to-sample reference distribution, $L(x) = \pi(x)/\pi_0(x)$ and the sequence $0 = \beta_0 < \beta_1 < \cdots < \beta_N = 1$ defines the *annealing schedule*. This bridge of auxiliary distributions is used to progressively transform samples from the *reference distribution* ($\beta = 0$) into samples from the *target distribution* ($\beta = 1$), for which only poorly mixing MCMC kernels may be available. For example, in the Bayesian setting where the target distribution is the posterior, we can choose the reference distribution as the prior, which we can often directly sample.

More precisely PT algorithms are based on Markov chains in which the states are $(N + 1)$-tuples, $\mathbf{x} = (x^0, x^1, x^2, \ldots, x^N) \in \mathcal{X}^{N+1}$, and whose stationary distribution is given by $\boldsymbol{\pi}(\mathbf{x}) = \prod_{i=0}^{N} \pi^{(\beta_i)}(x^i)$. At each iteration, PT proceeds by applying in parallel $N + 1$ MCMC kernels targeting $\pi^{(\beta_i)}$ for $i = 0, ..., N$. We call these model-specific kernels the *exploration kernels*. The chains closer to the reference chain (i.e. those with annealing parameter β_i close to zero) can

typically traverse regions of low probability mass under π while the chain at $\beta = 1$ ensures that asymptotically we obtain samples from the target distribution. Frequent communication between the chains at the two ends of the spectrum is therefore critical for good performance, and achieved by proposing to swap the states of chains at adjacent annealing parameters. *Even swap* moves (rows labelled 'E'), respectively *Odd swap* moves (labelled 'O'), propose to exchange states at chains with an even index i, resp. odd index i, and $i + 1$. These proposals are accepted or rejected according to a Metropolis mechanism.

1.2 Deterministic and stochastic even-odd schemes

The effectiveness of PT is partly determined by how quickly the swapping scheme can transfer information from the reference chain to the target chain. We can visualize this information transfer by monitoring the $N + 1$ index processes taking values in $\{0, 1, ..., N\}$ highlighted by the bold trajectories in Figure 1 and formally defined in Section 2.4. Each process is initialized at a distinct $i \in \{0, ..., N\}$ and tracks how the state of the corresponding chain evolves over the annealing schedule thanks to the swap move. When PT algorithms are distributed over several machines, it is critical that instead of having pairs of machines exchanging high-dimensional states when a swap is accepted (which could be detrimental due to network latency), the machines should just exchange the annealing parameters. In this case, the index process initialized at i is thus directly available to the machine where the i^{th}-chain was initialized.

There have been many proposals made to improve this information transfer by adjusting the annealing schedule; see, e.g., [Kone and Kofke, 2005, Atchadé et al., 2011, Miasojedow et al., 2013]. These proposals are useful but do not address a crucial limitation of standard PT algorithms. In a distributed context, one can select randomly at each iteration whether to apply Even or Odd swap moves in parallel. The resulting stochastic even-odd swap (SEO) scheme, henceforth referred to as *reversible PT* as it admits a reversible scaling limit (see Section 6), yields index processes exhibiting a diffusive behaviour. This is illustrated in the top row of Figure 1. Hence we can expect that when N is large it takes roughly $O(N^2)$ swap attempts for a state at $\beta_0 = 0$ to reach $\beta_N = 1$ [Diaconis et al., 2000]. The user thus faces a trade-off. If N is too large, the acceptance probabilities of the swap moves are high but it takes a time of order $O(N^2)$ for a state at $\beta = 0$ to reach $\beta = 1$. If N is too low, the acceptance probabilities of swap moves deteriorate resulting in poor mixing between the different chains. Informally, even in a multi-core or distributed setting, for N large, the $O(N)$ gains in being able to harness more cores do not offset the $O(N^2)$ cost of the diffusion (see Figures 3, and Section 3.5 where we formalize this argument). As a consequence, the general consensus is that the temperatures should be chosen to allow for about a 20–40% acceptance rate to maximize the square jump distance travelled per swap in the space of annealing parameters $[0, 1]$ [Kone and Kofke, 2005, Lingenheil et al., 2009, Atchadé et al., 2011]. Adding more chains past this threshold actually deteriorates the performance of reversible PT and there have even been attempts to adaptively reduce the number of additional chains [Lacki and Miasojedow, 2016]. This is a lost opportunity, since PT is otherwise particularly suitable to implementation on multi-core

or distributed architectures.

An alternative to the SEO scheme is the deterministic even-odd swap (DEO) scheme introduced in [Okabe et al., 2001] where one deterministically alternates Even and Odd swap moves. We refer to DEO as *non-reversible PT* as it admits a non-reversible scaling limit (see Section 6). In particular, the resulting index processes do not appear to exhibit a diffusive behaviour, illustrated by the bottom row of Figure 1. In fact, the communication between chains actually improves with N which leads to drastically different optimal tuning recommendations compared to reversible PT (see Section 5). This non-diffusive scaling behaviour of non-reversible PT explains its excellent empirical performance when compared to alternative PT schemes even with the (sub-optimal) tuning recommendations of reversible PT used in practice [Lingenheil et al., 2009].

1.3 Overview of our contributions

Previous theoretical studies analyzed the asymptotic behaviour of standard PT based on a target consisting of a product of independent components of increasing dimension [Atchadé et al., 2011], or an increased swap frequency relative to a continuous time sampling process [Dupuis et al., 2012]. We perform here an asymptotic analysis relevant to modern computational architectures such as GPUs and distributed computing, where we take the number of parallel chains and cores to infinity. One advantage of our approach is that, in contrast to these previous analyses, we do not need to make assumptions on the structure of neither the target distribution as in [Atchadé et al., 2011] nor the exploration kernels as in [Dupuis et al., 2012].

After introducing formally the SEO and DEO schemes in Section 2, our first contribution is a non-asymptotic result showing that the non-reversible DEO scheme is guaranteed to outperform its reversible SEO counterpart. The notion of optimality we analyze is the *round trip rate*, which quantifies how often information from the reference distribution percolates to the target; see Section 3.

In Section 4 we show that for non-reversible PT the round trip rate asymptotically increases to an upper bound, in contrast to the reversible counterpart for which it decays to zero. In Section 5 we combine the analysis from Section 3 and Section 4 to develop practical guidelines for practitioners:

1. For challenging sampling problems one should use non-reversible PT (Algorithm 1) with at least as many chains as the number of cores available, which contrasts to reversible PT where this can be detrimental.

2. Use Algorithm 3 for an empirically effective and computationally efficient adaptive algorithm to estimate the optimal schedule.

In Section 6 we identify the scaling limit of the index processes for both reversible and non-reversible PT as the number of parallel chains goes to infinity. We show that this scaling limit is a piecewise-deterministic Markov process for non-reversible PT whereas it is a diffusion for reversible PT as suggested by the dynamics of the bold paths in Figure 1.

Finally in Section 7, we present a variety of experiments validating our theoretical analysis and novel methodology. The computer code to replicate the experiments is implemented in an open source probabilistic programming available at https://github.com/UBC-Stat-ML/blangSDK.

2 Setup and notation

2.1 Parallel tempering

Henceforth we will assume that the *target* and *reference* probability distributions π and π_0 on $\mathcal{X}$ admit strictly positive densities with respect to a common dominating measure $\mathrm{d}x$. We will also denote these densities somewhat abusively by π and π_0. It will be useful to define $V_0(x) = -\log \pi_0(x)$ and $V(x) = -\log L(x)$, where $L(x) = \pi(x)/\pi_0(x)$ is assumed finite for all $x \in \mathcal{X}$. Using this notation, the *annealed distribution* at an annealing parameter β is given by

$$\pi^{(\beta)}(x) = \frac{L(x)^\beta \pi_0(x)}{\mathcal{Z}(\beta)} = \frac{e^{-\beta V(x) - V_0(x)}}{\mathcal{Z}(\beta)}, \tag{1}$$

where $\mathcal{Z}(\beta) = \int_{\mathcal{X}} L(x)^\beta \pi_0(x) \mathrm{d}x$ is the corresponding normalizing constant.

We denote the *annealing schedule* by $0 = \beta_0 < \beta_1 < \cdots < \beta_N = 1$. In our asymptotic analysis we will view it as a partition $\mathcal{P} = \{\beta_0, \ldots, \beta_N\}$ of $[0, 1]$ with mesh-size $\|\mathcal{P}\| = \sup_i \{\beta_i - \beta_{i-1}\}$. Given an annealing schedule $\mathcal{P}$, let $\boldsymbol{\pi}(\mathbf{x}) = \prod_{i=0}^N \pi^{(\beta_i)}(x^i)$ be the joint density on the augmented space $\mathcal{X}^{N+1}$ targeted by PT.

We now define formally the Markov kernels corresponding to the reversible (SEO) and non-reversible (DEO) PT algorithms described informally in the introduction and illustrated in Figure 1. For both SEO and DEO, the overall $\boldsymbol{\pi}$-invariant Markov kernel $\mathbf{K}_n^{\mathrm{PT}}$ describing the algorithm is obtained by the composition of a communication kernel $\mathbf{K}_n^{\mathrm{comm}}$ and an exploration kernel $\mathbf{K}^{\mathrm{expl}}$,

$$\mathbf{K}_n^{\mathrm{PT}} = \mathbf{K}_n^{\mathrm{comm}} \mathbf{K}^{\mathrm{expl}}, \tag{2}$$

where $\mathbf{K}_n^{\mathrm{comm}} \mathbf{K}^{\mathrm{expl}}(\mathbf{x}, A) = \int \mathbf{K}_n^{\mathrm{comm}}(\mathbf{x}, \mathrm{d}\mathbf{x}') \mathbf{K}^{\mathrm{expl}}(\mathbf{x}', A)$. The difference between SEO and DEO is in the communication phase, namely $\mathbf{K}_n^{\mathrm{comm}} = \mathbf{K}^{\mathrm{SEO}}$ in the former case and $\mathbf{K}_n^{\mathrm{comm}} = \mathbf{K}_n^{\mathrm{DEO}}$ in the latter. Both communication kernels are detailed further.

2.2 Exploration kernels

The exploration kernels are defined in the same way for SEO and DEO. They are also model specific, so we assume we are given one $\pi^{(\beta_i)}$-invariant kernel $K^{(\beta_i)}$ for each annealing parameter $\beta_0, \beta_1, \ldots, \beta_N$. These can be based on Hamiltonian Monte Carlo, Metropolis–Hastings, Gibbs Sampling, Slice Sampling, etc. The exploration kernel of the reference chain can often be taken to be π_0, i.e. $K^{(0)}(x, A_0) = \pi_0(A_0)$. We construct the overall exploration kernel by applying the

annealing parameter specific kernels to each component independently from each other:

$$\mathbf{K}^{\text{expl}}(\mathbf{x}, A_0 \times A_1 \times \dots A_N) = \prod_{i=0}^{N} K^{(\beta_i)}(x^i, A_i). \tag{3}$$

2.3 Communication kernels.

Before defining the communication scheme, it will be useful to first construct its fundamental building block, the swap kernel $\mathbf{K}^{(i,j)}$. A swap kernel is a Metropolis–Hastings move with a deterministic proposal which consists of permuting two coordinates of a state vector. The proposed state is denoted

$$\mathbf{x}^{(i,j)} = (x^0, x^1, \dots, x^{i-1}, x^j, x^{i+1}, \dots, x^{j-1}, x^i, x^{j+1}, \dots, \dots, x^N). \tag{4}$$

The Metropolis–Hastings kernel $\mathbf{K}^{(i,j)}$ corresponding to this update is given by

$$\mathbf{K}^{(i,j)}(\mathbf{x}, A) = (1 - \alpha^{(i,j)}(\mathbf{x}))\delta_{\mathbf{x}}(A) + \alpha^{(i,j)}(\mathbf{x})\delta_{\mathbf{x}^{(i,j)}}(A), \tag{5}$$

where δ_x denotes the Dirac delta and $\alpha^{(i,j)}(\mathbf{x})$ is the corresponding acceptance probability equal to

$$\alpha^{(i,j)}(\mathbf{x}) = \min\left\{1, \frac{\boldsymbol{\pi}\left(\mathbf{x}^{(i,j)}\right)}{\boldsymbol{\pi}(\mathbf{x})}\right\} \tag{6}$$

$$= \exp\left(\min\{0, (\beta_j - \beta_i)(V(x^j) - V(x^i))\}\right). \tag{7}$$

We define the even and odd kernels $\mathbf{K}^{\text{even}}$ and $\mathbf{K}^{\text{odd}}$,

$$\mathbf{K}^{\text{even}} = \prod_{i \text{ even}} \mathbf{K}^{(i,i+1)}, \qquad \mathbf{K}^{\text{odd}} = \prod_{i \text{ odd}} \mathbf{K}^{(i,i+1)}. \tag{8}$$

These kernels are maximal groups of swap moves such that members of the group do not interfere with each other.

For SEO, the kernel $\mathbf{K}_n^{\text{comm}} = \mathbf{K}^{\text{SEO}}$ is given by a mixture of the odd and even kernels in equal proportion:

$$\mathbf{K}^{\text{SEO}} = \frac{1}{2}\mathbf{K}^{\text{odd}} + \frac{1}{2}\mathbf{K}^{\text{even}}. \tag{9}$$

For DEO, the kernel $\mathbf{K}_n^{\text{comm}} = \mathbf{K}_n^{\text{DEO}}$ is given by a deterministic alternation between odd and even kernels. This is encoded by the following time heterogeneous kernel

$$\mathbf{K}_n^{\text{DEO}} = \begin{cases} \mathbf{K}^{\text{even}} & \text{if } n \text{ is even,} \\ \mathbf{K}^{\text{odd}} & \text{if } n \text{ is odd.} \end{cases} \tag{10}$$

We provide pseudo-code for the DEO scheme in Algorithm 1. As presented here, this algorithm also estimates the average rejection probabilities $r^{(i,i+1)}$ of swap moves between chains $(i, i+1)$

which are used to optimize the annealing schedule in Section 5. When the schedule is fixed, lines 1, 13, 18 can be omitted and we should use "for $i \in S$" on line 11.

Algorithm 1 Non-reversible PT (DEO)(number of scans n, annealing schedule $\mathcal{P}$)

1: $\hat{r}^{(i,i+1)} \leftarrow 0$ for all $i \in \{0, 1, \ldots, N-1\}$ ▷ Swap rejection statistics used in Section 5.4 to adapt the schedule
2: $\mathbf{x} \leftarrow \mathbf{x}_0$ ▷ Initialize chain
3: **for** t **in** 1, 2, ..., n **do**
4: **for** k **in** $1, 2, \ldots, n_{\text{expl}}$ **do**
5: $\mathbf{x}' \sim \mathbf{K}^{\text{expl}}(\mathbf{x}, \cdot)$ ▷ Exploration phase (parallelizable)
6: $\mathbf{x} \leftarrow \mathbf{x}'$
7: **if** t is even **then** ▷ Non-reversibility inducing alternation
8: $S \leftarrow \{i : 0 \le i < N, i \text{ is even}\}$ ▷ Even subset of $\{0, \ldots, N-1\}$
9: **else**
10: $S \leftarrow \{i : 0 \le i < N, i \text{ is odd}\}$ ▷ Odd subset of $\{0, \ldots, N-1\}$
11: **for** i **in** $0, \ldots, N-1$ **do** ▷ Communication phase (parallelizable)
12: $\alpha \leftarrow \alpha^{(i,i+1)}(\mathbf{x})$
13: $\hat{r}^{(i,i+1)} \leftarrow \hat{r}^{(i,i+1)} + (1 - \alpha)$
14: $A \sim \text{Bern}(\alpha)$
15: **if** $i \in S$ **and** $A = 1$ **then**
16: $(x^i, x^{i+1}) \leftarrow (x^{i+1}, x^i)$ ▷ Equation (7).
17: $\mathbf{x}_t \leftarrow \mathbf{x}$
18: $\hat{r}^{(i,i+1)} \leftarrow \hat{r}^{(i,i+1)}/n$ for all $i \in \{0, 1, \ldots, N-1\}$ ▷ Equation (33)
19: **return** $(\mathbf{x}_1, \ldots, \mathbf{x}_n), (\hat{r}^{(0,1)}, \ldots, \hat{r}^{(N-1,N)})$

2.4 Index process, chains, replicas

Our analysis of PT algorithms will be based on the index process which has been informally introduced in the introduction.

Suppose we initialize machine i with annealing parameter $\beta_i \in \mathcal{P}$. Then after n scans, the annealing parameters are randomly permuted among the $N+1$ machines. Let $\text{Perm}(N)$ be the set of bijections on $\{0, \ldots, N\}$, then there is a $\sigma_n \in \text{Perm}(N)$ such that machine i is responsible for annealing parameter $\beta_{\sigma_n(i)} \in \mathcal{P}$ as we recall that the machines swap the annealing parameters and not the states. The process $(\mathbf{X}_n, \sigma_n)$ where $\mathbf{X}_n := (\mathbf{X}_n^0, ..., \mathbf{X}_n^N)$ is a Markov chain on the augmented state space $\mathcal{X}^{N+1} \times \text{Perm}(N)$ with invariant distribution $\bar{\boldsymbol{\pi}}(\mathbf{x}, \sigma) = \boldsymbol{\pi}(\mathbf{x})/(N+1)!$; see Appendix A for details.

Let $I_n^i = \sigma_n(i)$ be the sequence of the indices of the annealing parameters at iteration n on machine i and $\varepsilon_n^i \in \{-1, 1\}$ denote the direction that I_n^i will be proposed at iteration n. We have $\varepsilon_n^i = 1$ if I_n^i is proposed an increase and -1 otherwise. We define the *index process* for machine i as $Y_n^i = (I_n^i, \varepsilon_n^i) \in \{0, \ldots, N\} \times \{-1, 1\}$. The concept is best understood visually: refer to the bold piecewise linear paths illustrating I_n^{N-2} for $N = 8$ and I_n^{N-1} for $N = 30$ in Figure 1.

Finally, we will refer to the sequences X_n^i, and $X_n^{\sigma_n(i)}$ as the i-th *chain* and *replica* respectively.

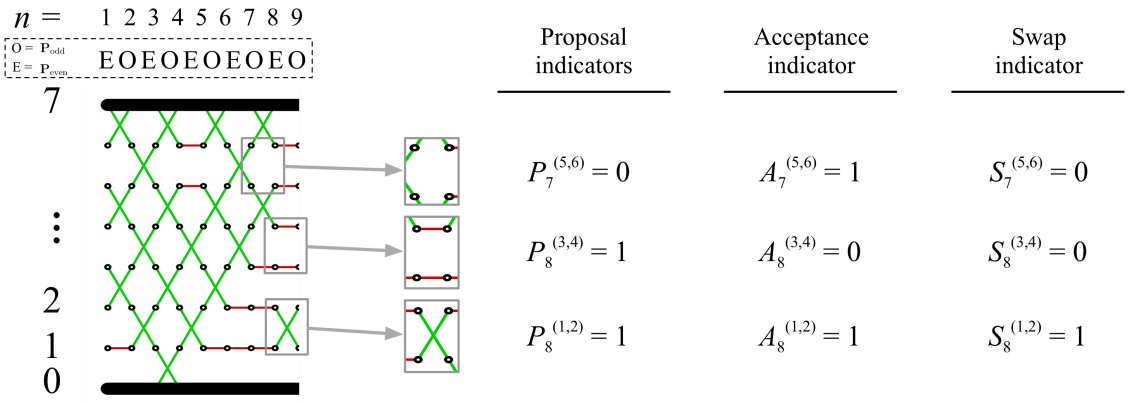

Figure 2: Illustration of the proposal, acceptance and swap indicators.

The i-th chain tracks the sequence of states with annealing parameter β_i, and the i-th replica tracks the sequence of states on machine i. We drop the index i when it is unimportant.

2.5 The index process for reversible and non-reversible PT

We will use the dynamics of the index process to explain the differences between SEO and DEO communication. Let $P_n^{(i,j)} \in \{0,1\}$ denotes an indicator that a swap is proposed between chains i and j at iteration n. The swap proposals are then defined from the proposal indicators as $S_n^{(i,j)} = P_n^{(i,j)} A_n^{(i,j)}$, where $A_n^{(i,j)} | \mathbf{X}_n \sim \text{Bern}(\alpha^{(i,j)}(\mathbf{X}_n))$ are acceptance indicator variables (see Figure 2). The index process $Y_n^i = (I_n^i, \varepsilon_n^i)$ satisfies the following recursive relation: initialize $Y_0^i = (i, 1)$ if $P_0^{(i,i+1)} = 1$ and $Y_0^i = (i, -1)$ otherwise. For $n > 0$, we have

$$I_{n+1}^i = \begin{cases} I_n^i + \varepsilon_n^i & \text{if } S_n^{(I_n^i, I_n^i + \varepsilon_n^i)} = 1, \\ I_n^i & \text{otherwise,} \end{cases}, \qquad \varepsilon_{n+1}^i = \begin{cases} 1 & \text{if } P_n^{(I_{n+1}^i, I_{n+1}^i + 1)} = 1, \\ -1 & \text{otherwise.} \end{cases} \tag{11}$$

The only difference between SEO and DEO is in the proposal indicators. Define $\mathbf{P}_n = (P_n^{(0,1)}, P_n^{(1,2)}, \dots, P_n^{(N-1,N)})$, $\mathbf{P}_n$ is deterministic for DEO, i.e. $\mathbf{P}_n = \mathbf{P}_{\text{even}} = (1, 0, 1, \dots)$ for even n and $\mathbf{P}_n = \mathbf{P}_{\text{odd}} = (0, 1, 0, \dots)$ for odd n. In SEO, we have $\mathbf{P}_n \sim \text{Unif}\{\mathbf{P}_{\text{even}}, \mathbf{P}_{\text{odd}}\}$.

For SEO, the variables $\varepsilon_n^i \sim \text{Uniform}\{-1, 1\}$ are i.i.d, and consequently the index process exhibits a random walk behaviour. In contrast for DEO, we have $\varepsilon_{n+1}^i = \varepsilon_n^i$ so long as $I_{n+1}^i = I_n^i + \varepsilon_n^i$ and $\varepsilon_{n+1}^i = -\varepsilon_n^i$ otherwise. Therefore the index process for DEO performs a more systematic exploration of the space as the direction ε_{n+1}^i is only reversed when a swap involving machine i is rejected or if the boundary is reached. The qualitative differences between the two regimes can be seen in Figures 1 and 3. In particular the index process for DEO in Figure 3 behaves very differently as N increases, this will be explored formally in Section 6.

As mentioned in the introduction, we refer to the PT algorithm with SEO and DEO communication as *reversible* PT and *non-reversible* PT respectively. Our terminology is somewhat abusive but is justified by the analysis in Sections 3.4 and 6, where it is shown that, under certain

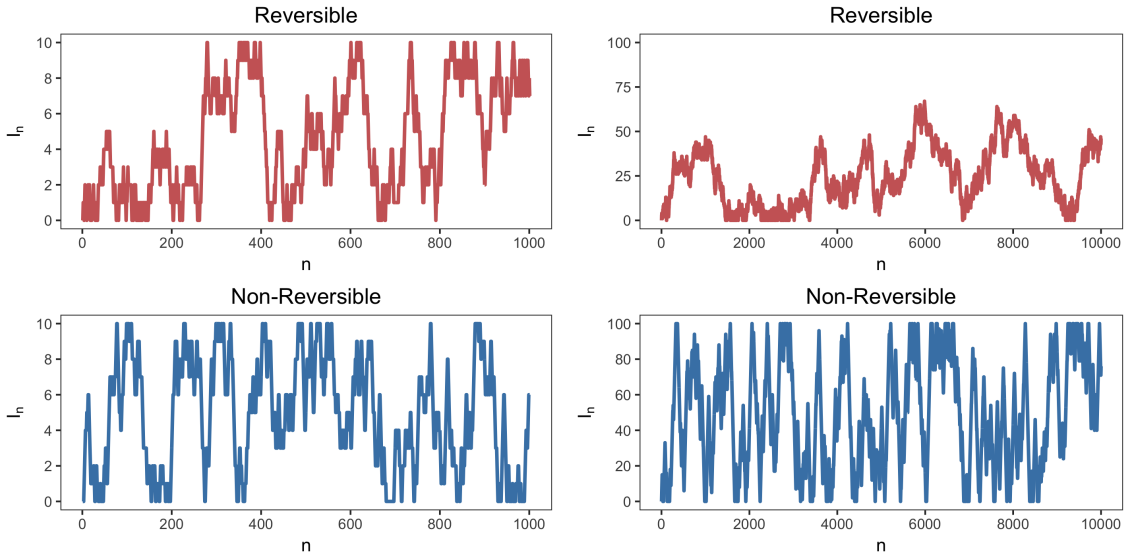

Figure 3: Sample trajectories of the component I_n^0 of the index process for a Gaussian model with $\Lambda = 5$ (see Section 7.1 for details) with schedule $\mathcal{P}$ satisfying (37) for $N = 10$ (left) and $N = 100$ (right). The trajectories are run over $100N$ iterations for reversible (top) and non-reversible (bottom) PT.

assumptions, the index process is reversible for SEO while it is non-reversible for DEO.

3 Non-asymptotic analysis of PT algorithms

3.1 Model of compute time

We start with a definition of what we model as one unit of compute time: throughout the paper, we assume a massively parallel or distributed computational setup, and hence that sampling once from each of the kernels $\mathbf{K}^{\text{expl}}$, $\mathbf{K}_n^{\text{DEO}}$ and $\mathbf{K}^{\text{SEO}}$ takes one unit of time, independently of the number of chains $N+1$. This assumption is realistic in both GPU and parallel computing scenarios, since the communication cost for each swap does not increase with the dimensionality of the problem. We also assume that the number of MCMC iterations will still dominate the number of parallel cores available, i.e. $n \gg N$. This is reasonable when addressing challenging sampling problems.

There are numerous empirical studies on multi-core and distributed implementation of PT [Altekar et al., 2004, Mingas and Bouganis, 2012, Fang et al., 2014]. However, we are not aware of previous theoretical work investigating such a computational model.

3.2 Performance metrics for PT methods

The standard notion of computational efficiency of MCMC schemes is the effective sample size (ESS) per compute time. However, for PT methods, since the ESS per compute time depends on the details of the problem specific exploration kernels $\mathbf{K}^{\text{expl}}$, alternatives have been developed in the literature to assess the performance of $\mathbf{K}_n^{\text{comm}}$ which are independent of $\mathbf{K}^{\text{expl}}$.

We are motivated by the Bayesian context where the reference chain ($\beta = 0$) and provides one independent sample at each iteration. In this context, notice that each generated sample from the reference chain has a chance to be propagated to the target chain ($\beta = 1$), we call this an *annealed restart*. Informally an annealed restart can be thought of as a sampling equivalent to what is known in optimization as a random restart. We say a *round trip* has occurred for replica i when the component $(I_n^i)_{n\geq 0}$ of its corresponding index process successfully increases from 0 to N and then goes back to 0.

Formally, we recursively define $T_{\downarrow,0}^i = \inf\{n : (I_n^i, \varepsilon_n^i) = (0, -1)\}$ and for $k \geq 1$,

$$T_{\uparrow,k}^i = \inf\{n > T_{\downarrow,k-1}^i : (I_n^i, \varepsilon_n^i) = (N, 1)\}, \tag{12}$$

$$T_{\downarrow,k}^i = \inf\{n > T_{\uparrow,k}^i : (I_n^i, \varepsilon_n^i) = (0, -1)\}. \tag{13}$$

The k-th annealed restart and round trip for replica i occurs at scan $T_{\uparrow,k}^i$ and $T_{\downarrow,k}^i$ respectively. Let $\mathcal{T}_n$ and $\mathcal{R}_n$ be the total number of annealed restarts and round trips respectively during the first n scans of Algorithm 1.

We wish to optimize for the percentage of iterations that result in an annealed restart, i.e. $\tau = \lim_{n\to\infty} \mathbb{E}[\mathcal{T}_n]/n$, where we use abusively the same random variables for SEO and DEO but differentiate these schemes by using the probability measures $\mathbb{P}_{\text{SEO}}$ and $\mathbb{P}_{\text{DEO}}$ with associated expectation operators $\mathbb{E}_{\text{SEO}}$ and $\mathbb{E}_{\text{DEO}}$. We use $\mathbb{P}$ and $\mathbb{E}$ for statements that hold for both algorithms. If $\mathcal{T}_n^i$ and $\mathcal{R}_n^i$ are the total number of annealed restarts and round trips during the first n iterations by replica i respectively, then we have $\mathcal{R}_n^i \leq \mathcal{T}_n^i \leq \mathcal{R}_n^i + 1$. Consequently, $\mathcal{R}_n \leq \mathcal{T}_n \leq \mathcal{R}_n + N + 1$ and thus $\tau = \lim_{n\to\infty} \mathbb{E}[\mathcal{R}_n]/n$.

In the PT literature, τ is commonly referred to as the *round trip rate* and has been used to compare the effectiveness of various PT algorithms [Katzgraber et al., 2006, Lingenheil et al., 2009].

Another performance metric commonly used in the PT literature is the *expected square jump distance* (ESJD) [Atchadé et al., 2011, Kone and Kofke, 2005], defined as

$$\text{ESJD} = \mathbb{E}\left[(\beta_{I+1} - \beta_I)^2 \, \alpha^{(I,I+1)}(\mathbf{X})\right], \tag{14}$$

where $I \sim \text{Unif}\{0, 1, 2, \ldots, N\}$ and $\mathbf{X} \sim \boldsymbol{\pi}$. While this criterion is useful within the context of reversible PT for selecting the optimal number of parallel chains, it is too coarse to compare reversible to non-reversible PT methods as, for any given annealing schedule, the ESJD is identical in both cases.

3.3 Model assumptions

The analysis of the round trip times is in general intractable because the index process Y_n is not Markovian. Indeed, simulating a transition depends on the acceptance indicators $A_n^{(i,i+1)}$ (see Section 2.4), the distributions of which themselves depend on the state configuration $\mathbf{X}$. To simplify the analysis, we will make in the remainder of the paper the following simplifying assumptions:

(A1) *Stationarity*: $\mathbf{X}_0 \sim \boldsymbol{\pi}$ and thus $\mathbf{X}_n \sim \boldsymbol{\pi}$ for all n as the kernel $\mathbf{K}_n^{\mathrm{PT}}$ is $\boldsymbol{\pi}$-invariant.

(A2) *Efficient Local Exploration (ELE)*: For $X \sim \pi^{(\beta)}$ and $X'|X \sim K^{(\beta)}(X, \cdot)$, the random variables $V(X)$ and $V(X')$ are independent.

(A3) *Integrability*: V^3 is integrable with respect to π_0 and π.

It follows from Assumption (A1) that the marginal behaviour of the communication scheme only depends on the distribution of the state $\mathbf{X}_n$ via the $N+1$ univariate distributions of the chain-specific energies $V^{(i)} = V\left(X^{(i)}\right)$, $i \in \{0, 1, 2, \ldots, N\}$. This follows from (7) as the acceptance probability $\alpha^{(i,i+1)}(\mathbf{X})$ only depends on $V^{(i)}, V^{(i+1)}$. This allows us to build a theoretical analysis which makes no structural assumption on the state space $\mathcal{X}$ as typically done in the literature [Atchadé et al., 2011,Roberts and Rosenthal, 2014]. In contrast, previous work such as [Atchadé et al., 2011] assume a product space $\mathcal{X} = \mathcal{X}_0^d$ for large d.

Assumption (A2) is weaker than assuming that X and X' are independent. Consider for example a scenario where we seek to explore the target distribution of a mixture model with symmetries induced by label switching. In such cases, being able to design exploration kernels such as $V(X)$ and $V(X')$ are approximately independent can be understood as being able to efficiently visit a neighbourhood of one of the local maxima. In contrast, being able to sample X' independently from X would defy the need for using PT in the first place.

In particular Assumptions (A1)-(A2) allow us to express the acceptance indicators as *independent* Bernoulli random variables $A_n^{(i,i+1)} \sim \mathrm{Bern}(s^{(i,i+1)})$ where $s^{(i,i+1)}$ is given by the expectation of Equation (7),

$$s^{(i,i+1)} = \mathbb{E}\left[\alpha^{(i,i+1)}(\mathbf{X})\right] = \mathbb{E}\left[\exp\left(\min\left\{0, (\beta_{i+1} - \beta_i)\left(V^{(i+1)} - V^{(i)}\right)\right\}\right)\right], \tag{15}$$

the expectation being over two independent random variables $V^{(i)}, V^{(i+1)}$, satisfying $V^{(i)} \overset{d}{=} V(X^{(\beta_i)})$ for $X^{(\beta_i)} \sim \pi^{(\beta_i)}$.

Even though these assumptions are not satisfied in real problems, they provide the foundations of a model for PT algorithms. We validate empirically the predictions made by the model in Section 7 and demonstrate its robustness even when the ELE assumption is severely violated.

3.4 Reversibility and non-reversibility of the index process

Under assumptions (A1)-(A3), for all i the index processes $Y_n^i = (I_n^i, \varepsilon_n^i)$ are homogeneous Markov processes with transition kernels P^{SEO} and P^{DEO} for reversible and non-reversible PT respectively. See Appendix B for an explicit representation of these kernels. We drop the superscript i and refer to $Y_n = (I_n, \varepsilon_n)$ as the index process.

The kernel P^{SEO} defines a reversible Markov chain on $\{0, \ldots, N\} \times \{-1, 1\}$ with uniform stationary distribution while P^{DEO} satisfies the skew-detailed balance condition w.r.t. to the same distribution,

$$P^{\mathrm{DEO}}(y, y') = P^{\mathrm{DEO}}(R(y'), R(y)), \tag{16}$$

where $R(i, \varepsilon) = (i, -\varepsilon)$ and is thus non-reversible. It falls within the generalized Metropolis–Hastings framework, see, e.g., [Lelièvre et al., 2010].

Reversibility necessitates that the Markov chain must be allowed to backtrack its movements. This leads to inefficient exploration of the state space. As a consequence, non-reversibility is typically a favourable property for MCMC chains. A common recipe to design non-reversible sampling algorithms consists of expanding the state space to include a "lifting" parameter that allows for a more systematic exploration of the state space [Chen et al., 1999, Diaconis et al., 2000, Turitsyn et al., 2011, Vucelja, 2016].

The index process $Y_n = (I_n, \varepsilon_n)$ for non-reversible PT can be interpreted as a "lifted" version of the index process for reversible PT with lifting parameter ε_n. Under DEO communication, I_n travels in the direction ε_n and only reverses direction when I_n reaches a boundary or when a swap rejection occurs. This "lifting" construction helps explain the qualitatively different behaviour between reversible and non-reversible PT and will be further explored when identifying the scaling limit of Y_n in Section 6. The lifted PT of [Wu, 2017] exploits a similar construction but only one of the $N + 1$ index processes is lifted instead of all of them for DEO. A lifted version of simulated tempering has also been proposed by [Sakai and Hukushima, 2016].

3.5 Non-asymptotic domination of non-reversible PT

Assumptions (A1)-(A3) ensure that for each $i = 0, \ldots, N$, $\mathcal{R}_n^i$ is a delayed renewal processes with round trip times $T_k^i = T_{\downarrow,k}^i - T_{\downarrow,k-1}^i$ for $k \geq 1$ and $i = 0, \ldots, N$. In particular, T_k^i are independent and identically distributed and, for convenience, we introduce the random variable $T \overset{d}{=} T_k^i$. By the key renewal theorem,

$$\tau = \sum_{i=0}^{N} \lim_{n \to \infty} \frac{\mathbb{E}[\mathcal{R}_n^i]}{n} = \frac{N+1}{\mathbb{E}[T]}. \tag{17}$$

An analytical expression for $\mathbb{E}[T]$ for reversible PT was first derived by [Nadler and Hansmann, 2007]. We extend this result to non-reversible PT in Theorem 1.

Theorem 1. *For any annealing schedule $\mathcal{P} = \{\beta_0, \ldots, \beta_N\}$,*

$$\mathbb{E}_{\mathrm{SEO}}[T] = 2(N+1)N + 2(N+1)E(\mathcal{P}), \tag{18}$$

$$\mathbb{E}_{\mathrm{DEO}}[T] = 2(N+1) + 2(N+1)E(\mathcal{P}), \tag{19}$$

where $E(\mathcal{P}) = \sum_{i=1}^{N} r^{(i-1,i)}/s^{(i-1,i)}$, and $r^{(i-1,i)} = 1 - s^{(i-1,i)}$ is the probability of rejecting a swap between chains i and $i+1$.

The proof can be found in Appendix C. Intuitively, Theorem 1 implies $\mathbb{E}[T]$ can be decomposed as the independent influence of communication scheme $\mathbf{K}_n^{\mathrm{comm}}$ and schedule $\mathcal{P}$ respectively. When all proposed swaps are accepted, the index process for reversible PT reduces to a simple random walk on $\{0, \ldots, N\}$, whereas for non-reversible PT, the index processes takes a direct path from

0 to N and back. Therefore, the first term in (18) and (19) represents the expected time for a round trip to occur in this idealized, rejection-free setting. The second term of (18) and (19) are identical and represent the additional time required to account for rejected swaps under schedule $\mathcal{P}$. Motivated by Theorem 1, we will refer to $E(\mathcal{P})$ as the *schedule inefficiency*.

By applying Theorem 1 to Equation (17), we get a non-asymptotic formula for the round trip rate in terms of $E(\mathcal{P})$.

Corollary 1. *For any annealing schedule $\mathcal{P}$ we have*

$$\tau_{\mathrm{SEO}}(\mathcal{P}) = \frac{N+1}{\mathbb{E}_{\mathrm{SEO}}[T]} = \frac{1}{2N + 2E(\mathcal{P})}, \tag{20}$$

$$\tau_{\mathrm{DEO}}(\mathcal{P}) = \frac{N+1}{\mathbb{E}_{\mathrm{DEO}}[T]} = \frac{1}{2 + 2E(\mathcal{P})}. \tag{21}$$

Consequently, $\tau_{\mathrm{SEO}}(\mathcal{P}) < \tau_{\mathrm{DEO}}(\mathcal{P})$ for $N > 1$.

4 Asymptotic analysis of PT

4.1 The communication barrier

We begin by analyzing the behaviour of the PT swaps as $\|\mathcal{P}\|$ goes to zero. In order to do so, we define the swap and rejection functions $s, r : [0,1]^2 \to [0,1]$ respectively as,

$$s(\beta, \beta') = \mathbb{E}\left[\exp\left(\min\{0, (\beta' - \beta)(V^{(\beta')} - V^{(\beta)})\}\right)\right], \tag{22}$$

$$r(\beta, \beta') = 1 - s(\beta, \beta'), \tag{23}$$

where $V^{(\beta)} \overset{d}{=} V(X^{(\beta)})$ for $X^{(\beta)} \sim \pi^{(\beta)}$ and $V^{(\beta)}, V^{(\beta')}$ are independent. The quantities $s(\beta, \beta')$ and $r(\beta, \beta')$ are symmetric in their arguments and represent the probability of swap and rejection occurring between β and β' respectively under the ELE assumption (A2). Note that $s^{(i-1,i)} = s(\beta_{i-1}, \beta_i)$.

To take the limit as $\|\mathcal{P}\| \to 0$, it will be useful to understand the behaviour of $r(\beta, \beta')$ when $\beta \approx \beta'$. The key quantity that drives this asymptotic regime is given by a function $\lambda : [0,1] \to [0, \infty)$ defined as the instantaneous rate of rejection of a proposed swap at annealing parameter β,

$$\lambda(\beta) = \lim_{\delta \to 0} \frac{r(\beta, \beta + \delta) - r(\beta, \beta)}{|\delta|}. \tag{24}$$

We define its integral by $\Lambda(\beta) = \int_0^\beta \lambda(\beta')\mathrm{d}\beta'$ and denote $\Lambda = \Lambda(1)$. Extending Proposition 1 in [Predescu et al., 2004] provides the following result.

Theorem 2. *λ is twice continuously differentiable and is equal to*

$$\lambda(\beta) = \frac{1}{2}\mathbb{E}\left[|V_1^{(\beta)} - V_2^{(\beta)}|\right], \tag{25}$$

where $V_1^{(\beta)}, V_2^{(\beta)}$ *are independent random variables with common distribution* $V^{(\beta)}$*. Moreover, we have*

$$r(\beta, \beta') = |\Lambda(\beta') - \Lambda(\beta)| + O(|\beta' - \beta|^3). \tag{26}$$

See Appendix **??** for a proof and general smoothness properties of λ. In particular, the existence and smoothness of λ is guaranteed by assumption (A3).

Theorem 2 shows that λ encodes up to second order the behaviour of r as the annealing parameter difference between the chains goes to 0. When $\lambda(\beta)$ is high, swaps are much more likely to be rejected, implying $\lambda(\beta)$ measures the difficulty of local communication for a chain with annealing parameter β.

Notice that $\Lambda \geq 0$ with equality if and only if $\lambda(\beta) = 0$ for all $\beta \in [0, 1]$. It can be easily verified from (25) that $\lambda = 0$ if and only if $V^{(\beta)}$ is constant $\pi^{(\beta)}$-a.s. for all $\beta \in [0, 1]$ which happens precisely when $\pi_0 = \pi$. So Λ defines a natural symmetric divergence measuring the difficulty of communication between π_0 and π. In particular, for any schedule $\mathcal{P}$, the sum of the rejection rates is approximately constant and equal to Λ as formalized in Corollary 2.

Corollary 2. *For any schedule* $\mathcal{P}$*,*

$$\sum_{i=1}^{N} r(\beta_{i-1}, \beta_i) = \Lambda + O(N\|\mathcal{P}\|^3). \tag{27}$$

Motivated by Theorem 2 and Corollary 2 we will henceforth refer to λ and Λ as the *local* and *global communication barrier* respectively.

4.2 High-dimensional scaling of communication barrier.

We determine here the asymptotic behaviour of λ and Λ when the dimension of $\mathcal{X}$ is large. To make the analysis tractable, we assume that $\pi_d(x) = \prod_{i=1}^{d} \pi(x_i)$ as in [Atchadé et al., 2011, Roberts and Rosenthal, 2014]. This provides a model for weakly dependent high-dimensional distributions. We only make this structural assumption on the state space and distribution to establish Proposition 1 below.

The corresponding annealed distributions are thus given by

$$\pi_d^{(\beta)}(x) = \prod_{i=1}^{d} \pi^{(\beta)}(x_i) \propto \exp\left(-\beta \sum_{i=1}^{d} V(x_i) - \sum_{i=1}^{d} V_0(x_i)\right). \tag{28}$$

Let λ_d and Λ_d be the local and global communication barriers for π_d respectively.

Proposition 1 (High Dimensional Scaling)**.** *Define* $\sigma^2(\beta) = \mathrm{Var}(V^{(\beta)})$*, for all* $\beta \in [0, 1]$ *we have as* $d \to \infty$*,*

$$\lambda_d(\beta) \sim \sqrt{\frac{d}{\pi}}\sigma(\beta), \qquad \Lambda_d \sim \sqrt{\frac{d}{\pi}} \int_0^1 \sigma(\beta) \mathrm{d}\beta. \tag{29}$$

See Appendix D for a detailed proof. It follows from Proposition 1 that λ_d and Λ_d increase at a $O(d^{1/2})$ rate as $d \to \infty$.

4.3 Asymptotic analysis of round trip rate

Suppose $\mathcal{P}_N$ is a sequence of annealing schedules of size $N+1$ such that $\mathcal{P}_N \subset \mathcal{P}_{N+1}$. Theorem 3 characterize the behaviour of the round trip rate as $\|\mathcal{P}_N\| \to 0$ through the schedule inefficiency $E(\mathcal{P}_N)$. In order to state Theorem 3 we will need the following notion of convergence: we write $a_n \lesssim b_n$ as $n \to \infty$ if and only if there is c_n such that $a_n \leq c_n$ and $c_n \sim b_n$ as $n \to \infty$. We say a_n is *asymptotically decreasing*, respectively *asymptotically increasing*, to a at rate $O(\delta_n)$ if $a_{n+1} \lesssim a_n$, respectively $a_n \lesssim a_{n+1}$, and $|a_n - a| = O(\delta_n)$.

Theorem 3. *If $\|\mathcal{P}_N\| \to 0$, then:*

(a) $E(\mathcal{P}_N)$ asymptotically decreases to Λ at a $O(\|\mathcal{P}_N\|)$ rate.

(b) The round trip rate τ_{SEO} goes to zero:

$$\tau_{SEO}(\mathcal{P}_N) \sim \frac{1}{2N} \to 0. \tag{30}$$

(c) The round trip rate τ_{DEO} asymptotically increases at a $O(\|\mathcal{P}_N\|)$ rate to the following upper bound:

$$\tau_{DEO}(\mathcal{P}_N) \to \bar{\tau} = \frac{1}{2 + 2\Lambda} > 0. \tag{31}$$

See Appendix E for a proof. In general, Λ is large when π_0 deviates significantly from π. Since Λ is problem specific, this identifies a limitation of PT present even in its non-reversible flavour, namely that adding more cores to the task will never be harmful, but does have a diminishing return. The bound $\bar{\tau}$ could indeed be very small for complex problems. Moreover, it is independent of the choice of annealing schedule, hence it cannot be improved by the schedule optimization procedure described in Section 5.

5 Tuning non-reversible PT algorithms

5.1 Optimal round trip rate

We show here that for a fixed annealing schedule $\mathcal{P}$ of size $N+1$, the round trip rate $\tau_{\mathrm{DEO}}(\mathcal{P})$ is maximized when the swap acceptance probabilities are equal. To solve this optimization problem, we first use Corollary 1 to rewrite the maximization of the round trip rate $\tau_{\mathrm{DEO}}(\mathcal{P})$ into a minimization of the schedule inefficiency, $E(\mathcal{P}) = \sum_{i=1}^{N} r_i/(1-r_i)$ where $r_i = r(\beta_{i-1}, \beta_i)$.

To get a tractable approximate characterization of the feasible region of $r_1, r_2, \ldots, r_N$, we use Corollary 2, which implies $\sum_{i=1}^{N} r_i \approx \Lambda$ for all schedules $\mathcal{P}$. Therefore assuming $\|\mathcal{P}\|$ is small enough

to ignore the error term, finding $\mathcal{P}_{\text{optimal}}$ is approximately equivalent to minimizing $\sum_{i=1}^{N} r_i/(1-r_i)$, subject to the constraint $\sum_{i=1}^{N} r_i = \Lambda$ and $r_i > 0$. This can be done using Lagrange multipliers and leads to a solution where r_i^* is constant in i, which we denote by r^*.

This "equi-acceptance" result is not surprising. Other theoretical frameworks and notions of efficiency also recommend equal acceptance rate between chains [Atchadé et al., 2011, Lingenheil et al., 2009, Kofke, 2002, Predescu et al., 2004], however implementing this equi-acceptance recommendation in practice is non-trivial. Previous work relied on stochastic approximation schemes [Atchadé et al., 2011, Miasojedow et al., 2013].

5.2 Optimal annealing schedule

We have shown that, for a fixed N, we should target an equi-acceptance annealing schedule. However, algorithmically we need to perform the optimization over the annealing parameters β_i in order to be able to run PT. Assuming λ is known for the time being, its estimation being postponed to the next section, the idea is that to obtain the optimal schedule $\mathcal{P}_{\text{optimal}} = \{\beta_0^*, \dots, \beta_N^*\}$, we partition the interval $[0, 1]$ such that the area under the curve λ between successive β_i^* and β_{i+1}^* is constant and equal to r^*.

To formalize this intuition, recall that for the optimal schedule $\mathcal{P}_{\text{optimal}}$ of size $N+1$, we have $r_i = r^*$ for all i. Theorem 2 and Corollary 2 imply that r^* must satisfy, $r^* \approx \Lambda(\beta_i) - \Lambda(\beta_{i-1})$ for all i and $r^* \approx \Lambda/N$ with an $O(\|\mathcal{P}\|^3)$ error. By equating these two estimates and summing from $i = 0, \dots, k$ we get

$$\Lambda(\beta_k^*) \approx \Lambda \frac{k}{N}, \tag{32}$$

with an error of $O(N\|\mathcal{P}\|^3)$. If we ignore error terms, (32) implies that $\beta_k^* \approx G(k/N)$ where $G = F^{-1}$ and $F(\beta) = \Lambda(\beta)/\Lambda$. Therefore finding β_k^* is approximately equivalent to finding the k/N quantiles of a random variable with density proportional to λ.

5.3 Estimation of the communication barrier

Computing $\lambda(\beta)$ or $\Lambda(\beta)$ exactly via (25) is in general intractable. We present here a simple Monte Carlo approximation. Assume we have access to a collection of samples, $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ from a non-reversible PT scheme based on an arbitrary annealing schedule $\mathcal{P}_N$ of size $N+1$ (see Algorithm 1). These samples may come from a short pilot run, or, as described in the next section, from the previous iteration of an adaptive scheme. For a given schedule $\mathcal{P}$, when the central limit theorem for Markov chains holds, the Monte Carlo estimates for the rejection rates satisfy

$$\hat{r}^{(i-1,i)} = \frac{1}{n} \sum_{k=1}^{n} \alpha^{(i-1,i)}(\mathbf{X}_k) = r^{(i-1,i)} + O_p(n^{-1/2}). \tag{33}$$

Next, using Theorem 2 we obtain $\sum_{j=1}^{i} r^{(j-1,j)} = \Lambda(\beta_i) + O(N\|\mathcal{P}\|^3)$. This motivates the following approximation for $\Lambda(\beta_i)$,

$$\hat{\Lambda}(\beta_i) = \sum_{j=1}^{i} \hat{r}^{(j-1,j)}, \tag{34}$$

which has an error of $O_p(\sqrt{N/n} + N\|\mathcal{P}\|^3)$.

We also obtain a consistent estimator $\hat{\tau} = (2 + 2\hat{\Lambda})^{-1}$ for the optimal round trip rate $\bar{\tau}$, where $\hat{\Lambda} = \hat{\Lambda}(1)$. In particular $\hat{\tau}$ allows us to diagnose if a low round trip is due to design choices for PT, or due to π, π_0. We can compare the empirically observed round trip rate against $\hat{\tau}$ to determine how far our implementation deviates from optimal performance.

Given the estimates for $\Lambda(\beta_0), \ldots, \Lambda(\beta_N)$ obtained above, we estimate the function $\Lambda(\beta)$ via interpolation, with the constraint that the interpolated function should be monotone increasing since $\lambda(\beta) \geq 0$. Specifically, we use the Fritsch-Carlson monotone cubic spline method [Fritsch and Carlson, 1980] and denote the monotone interpolation by $\hat{\Lambda}(\beta)$.

While we only use $\Lambda(\beta)$ in our adaptation procedure, it is still useful to estimate $\lambda(\beta)$ for visualization purpose. We do this by taking the derivative of our interpolation, $\hat{\lambda}(\beta) = \hat{\Lambda}'(\beta)$, which is a piecewise quadratic function.

5.4 Adaptive algorithm

The ideas described in this section so far are summarized in Algorithm 2, which given rejection statistics collected for a fixed annealing schedule provides an updated schedule.

Algorithm 2 UpdateSchedule(swap rejection estimates $\{\hat{r}^{(i-1,i)}\}$, previous schedule $\mathcal{P}$)

$N \leftarrow |\mathcal{P}| - 1$
For each $\beta_i \in \mathcal{P}$, compute $\hat{\Lambda}(\beta_i)$ ▷ Equation (34)
$S \leftarrow \{(\beta_0, \hat{\Lambda}(\beta_0)), (\beta_1, \hat{\Lambda}(\beta_1)), \ldots, (\beta_N, \hat{\Lambda}(\beta_N))\}$
Compute a monotone increasing interpolation $\hat{\Lambda}(\cdot)$ of the points S ▷ e.g. using [Fritsch and Carlson, 1980]
$\hat{\Lambda} \leftarrow \hat{\Lambda}(1)$
for k in $1, 2, ..., N-1$ **do**
    Find β_k^* such that $\hat{\Lambda}(\beta_k^*) = \hat{\Lambda}\frac{k}{N}$ using e.g. bisection.
return $\mathcal{P}^* = (0, \beta_1^*, \beta_2^*, \ldots, \beta_{N-1}^*, 1)$

As shown in Algorithm 3, we can also further exploit this idea to iteratively refine the annealing schedule. Algorithm 3 is based on a tuning parameter $b \in \{2, 3, \ldots\}$ with the interpretation that the fraction of samples used for adaptation is approximately b^{-1} (we use $b = 2$ in all our experiments). The adaptive procedure is designed so that it is an *anytime inference algorithm* with respect to the number of rounds performed [Zilberstein, 1996], meaning that $\rho + 1$ rounds of Algorithm 3 can be performed by running one additional round started from the output of the execution for ρ rounds.

Algorithm 3 is qualitatively different from existing adaptive PT algorithms [Atchadé et al., 2011, Miasojedow et al., 2013, Lacki and Miasojedow, 2016] which rely on continuous adaptation.

Algorithm 3 Non-reversible PT with adaptation

$N+1 \leftarrow$ number of cores available
$\mathcal{P} \leftarrow$ initial annealing schedule of size $N+1$ (e.g. uniform)
$n \leftarrow 1$
for round **in** 1, 2, ..., number of rounds requested **do**
    $\{\hat{r}^{(i-1,i)}\} \leftarrow \text{DEO}(n, \mathcal{P})$ ▷ Algorithm 1
    $\mathcal{P} \leftarrow \text{UpdateSchedule}(\{\hat{r}^{(i-1,i)}\}, \mathcal{P})$ ▷ Algorithm 2
    $n \leftarrow bn$ ▷ Rounds use an exponentially increasing number of scans

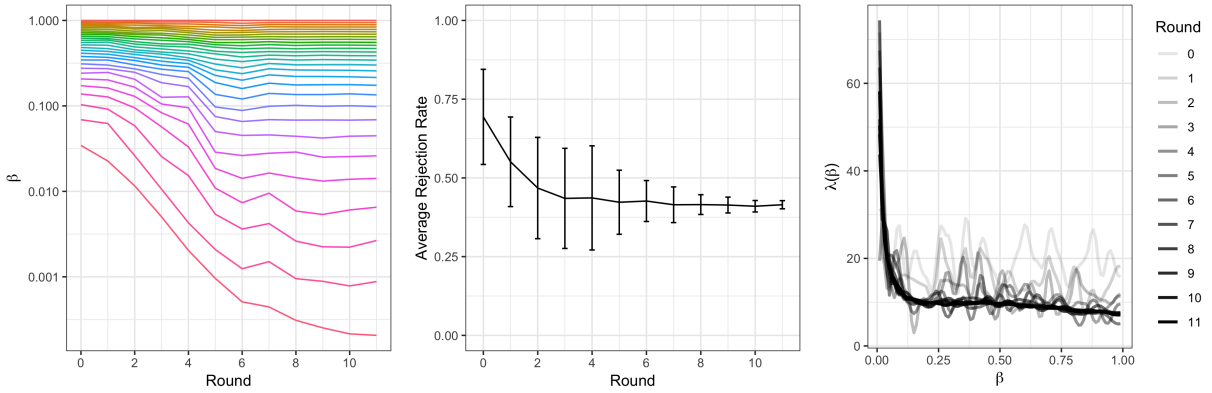

Figure 4: Algorithm 3 ran on a hierarchical Bayesian model applied to the historical failure rates of 5 667 launches for 367 types of rockets (a 369 dimensional problem, see Appendix G.4 for details). We use $N = 30$ chains and 11 adaptive rounds, the last one consisting of 5 000 scans and $\hat{\Lambda} = 12.03$. (Left) Progression of the adaptive annealing schedule (colours index parallel chains, y-axis, the values β_k for each adaptation round, in log scale). (Center) Progression of the sample mean and standard deviation of empirical rejection probabilities $\{\hat{r}^{(i-1,i)}\}_{i=1}^{N}$. The mean stabilizes quickly, but as the adaptive rounds increase, the rejection probabilities converge to the mean as desired. (Right) Progression of the estimated $\hat{\lambda}(\beta)$ evolution with adaption rounds.

Instead, we simply use here the last $1-b^{-1}$ fraction of the samples produced by Algorithm 3, which by construction follow an homogeneous chain. We found that Algorithm 3 experimentally outperforms existing adaptive methods in terms of round trip rates and ESS per second, as discussed in Section 7.4.

6 Scaling limits of index process

The DEO communication scheme introduced in [Okabe et al., 2001] was presumably devised on algorithmic grounds (it performs the maximum number of swap attempts in parallel) since no theoretical justification was provided and the non-reversibility of the scheme was not mentioned. The arguments given in [Lingenheil et al., 2009] to explain the superiority of DEO communication over various PT algorithms rely on an erroneous assumption, namely a diffusive scaling limit for the index process. Figure 3 suggests that the index process behaves qualitatively differently as N increases for reversible and non-reversible PT. The goal of this section is to investigate these differences by identifying some scaling limits. Such limits exist under assumptions (A1)-(A3) specified

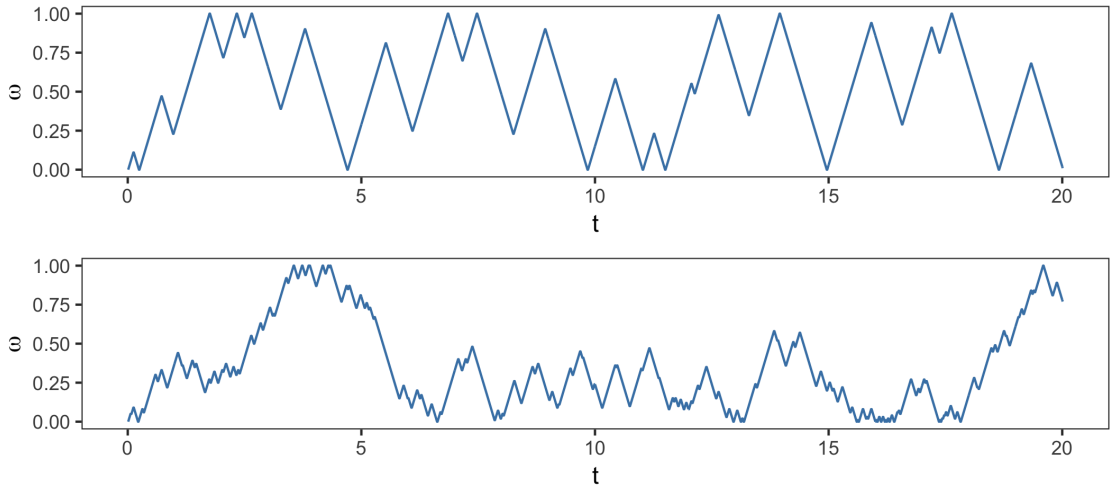

Figure 5: Sample trajectories of $W(t)$ where $Z(t) = (W(t), \varepsilon(t))$ under an optimal schedule generated by $G = F^{-1}$ for $F(\beta) = \Lambda(\beta)/\Lambda$ for $\Lambda = 1$ (top) and $\Lambda = 10$ (bottom) respectively.

in Section 3.3 and Section 4.1.

Suppose $G : [0,1] \to [0,1]$ is an increasing differentiable function satisfying $G(0) = 0$ and $G(1) = 1$. We say that G is a *schedule generator* for $\mathcal{P}_N = \{\beta_0, \ldots, \beta_N\}$ if $\beta_i = G(i/N)$. We will now assume without loss of generality that the sequence of schedules $\mathcal{P}_N$ are generated by some common G. In particular the mean value theorem implies $\|\mathcal{P}_N\| = O(N^{-1})$ as $N \to \infty$. This is not a strict requirement as most annealing schedules commonly used fall within this framework: the uniform schedule $\mathcal{P}_{\text{uniform}} = \{0, 1/N, \ldots, 1\}$ is generated by $G(w) = w$, the optimal schedule $\mathcal{P}_{\text{optimal}} = \{\beta_0^*, \ldots, \beta_N^*\}$ derived in Section 5.2 is approximately generated by $G(w) = F^{-1}(w)$, where $F(\beta) = \Lambda(\beta)/\Lambda$. If $\pi_0(x) \propto \pi(x)^\gamma$ for some $\gamma \in (0,1)$, and $L(x) \propto \pi(x)^{1-\gamma}$ then $G(w) = \frac{\gamma^{1-w}-\gamma}{1-\gamma}$ corresponds to the geometric schedule commonly used by practitioners.

Suppose $Y_n = (I_n, \varepsilon_n)$ is the index process for an annealing schedule $\mathcal{P}_N$ generated by G. To establish a scaling limit for Y_n, it will be convenient to work in a continuous time setting. To do this, we suppose the times that PT iterations occur are distributed according to a Poisson process $\{M(\cdot)\}$ with mean μ_N. The number $M(t)$ of PT iterations that occur by time $t \geq 0$ thus satisfies $M(t) \sim \text{Poisson}(\mu_N t)$. We define the *scaled index process* by $Z^N(t) = (W^N(t), \varepsilon^N(t))$ where $W^N(t) = I_{M(t)}/N$ and $\varepsilon^N(t) = \varepsilon_{M(t)}$.

Define the piecewise-deterministic Markov process (PDMP) $Z(t) = (W(t), \varepsilon(t))$ on $[0,1] \times \{-1, 1\}$ as follows: $W(t)$ moves in $[0,1]$ with velocity $\varepsilon(t)$ and the sign of $\varepsilon(t)$ is reversed at an inhomogeneous rate $\lambda(G(W(t))G'(W(t))$ or when $W(t)$ hits a boundary; see [Bierkens et al., 2018] for a discussion of PDMP on restricted domains. For the optimal schedule generated by $G = F^{-1}$ for $F(\beta) = \Lambda(\beta)/\Lambda$, we have $\lambda(G(w))G'(w) = \Lambda$ for all $w \in [0,1]$, so $\varepsilon(t)$ changes sign at a constant rate Λ. See Figure 5 for sample trajectories of Z for different values of Λ.

Theorem 4. *Suppose $\mathcal{P}_N$ is family of schedules generated by G, then*

(a) For reversible PT if $\mu_N = N^2$ and if $W^N(0)$ converges weakly to $W(0)$ then W^N converges weakly to a diffusion W, where W is a Brownian motion on $[0,1]$ with reflective boundary conditions. The process W admits $\mathrm{Unif}([0,1])$ *as stationary distribution.*

(b) For non-reversible PT if $\mu_N = N$ and if $Z^N(0)$ converges weakly to $Z(0)$, then Z^N converges weakly to the PDMP Z with initial condition $Z(0)$. The process Z admits $\mathrm{Unif}([0,1]\times\{-1,1\})$ *as stationary distribution.*

See Appendix F for a detailed construction of the scaled index process via their infinitesimal generators, and the proof of Theorem 4.

Theorem 4 implies for reversible PT that, if we speed time by a factor of N^2, then the index process scales to to a diffusion W independent of the choice of π_0, π and of the schedule. This is in contrast to non-reversible PT where, if we speed time by a factor of N, the index process converges to a PDMP Z depending on π_0, π through λ and the schedule through G.

7 Numerical experiments

7.1 Gaussian model

Suppose $\pi \sim N(0, \tau^{-1}\mathbb{I}_d)$, and $\pi_0 \sim N(0, \tau_0^{-1}\mathbb{I}_d)$ with $\tau_0 < \tau$. It can be shown that $\pi^{(\beta)} \sim N(0, \tau_\beta^{-1}\mathbb{I}_d)$ where $\tau_\beta = (1-\beta)\tau_0 + \beta\tau$. Theorem 1 in [Predescu et al., 2004] implies the following closed form expressions for $\lambda(\beta)$ and $\Lambda(\beta)$

$$\lambda(\beta) = \frac{2^{1-d}(\tau - \tau_0)}{B\left(\frac{d}{2}, \frac{d}{2}\right)\tau_\beta}, \qquad \Lambda(\beta) = \frac{2^{1-d}}{B\left(\frac{d}{2}, \frac{d}{2}\right)} \log\left(\frac{\tau_\beta}{\tau_0}\right), \tag{35}$$

where $B(a,b)$ is the Beta function. As $d \to \infty$, we have

$$\Lambda \sim \sqrt{\frac{d}{2\pi}} \log\left(\frac{\tau}{\tau_0}\right), \tag{36}$$

which is consistent with Proposition 1. We see from Figure 6 that the empirical approximation of $\hat{\lambda}, \hat{\Lambda}$ from Algorithm 3 are consistent with the theoretical values from in (35).

To determine the optimal annealing schedule $\mathcal{P}_{\text{optimal}} = \{\beta_0^*, \ldots, \beta_N^*\}$, we substitute (35) into in $\Lambda(\beta_k^*) = \Lambda\frac{k}{N}$ and solve for β_k^* as discuss in Section 5.2. This implies the optimal schedule satisfies

$$\tau_{\beta_k^*} = \tau_{\beta_{k-1}^*} \left(\frac{\tau}{\tau_0}\right)^{\frac{1}{N}}. \tag{37}$$

This is the same spacing obtained (based on a different theoretical approach) in [Atchadé et al., 2011] and [Predescu et al., 2004] for the Gaussian model.

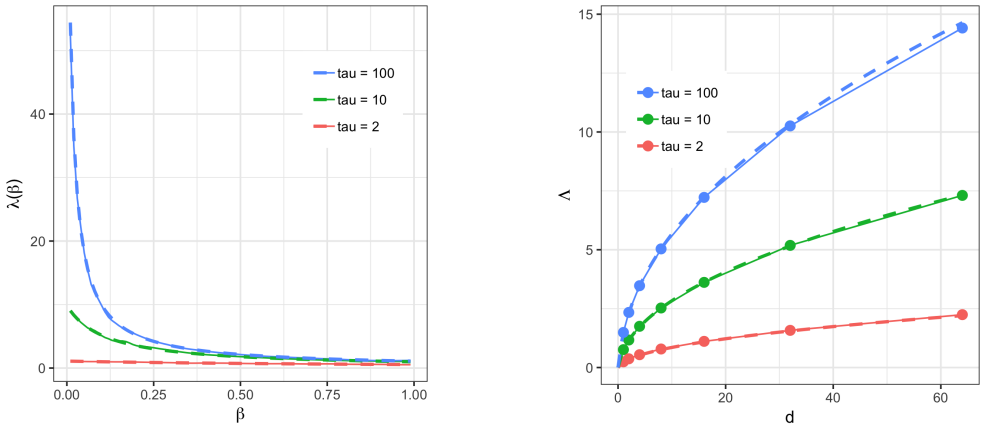

Figure 6: (Left) The local communication barrier for $d = 8, \tau = 2, 10, 100$ and $\tau_0 = 1$. (Right) The global communication barrier as a function of d for $\tau = 2, 10, 100$ and $\tau_0 = 1$. The solid line is the approximation $\hat{\lambda}(\beta)$ (respectively $\hat{\Lambda}$), resulting from Algorithm 3 ($N = 60$ and $n = 10\,000$ scans) and the dotted line is the analytic expression from (35).

7.2 Ising model

We now compute numerically λ for the Ising model on a 2-dimensional lattice of size $M \times M$ with magnetic moment μ. Using the notation $x_i \sim x_j$ to indicate sites are nearest neighbours on the lattice, the target distribution is annealed by the inverse temperature β and the tempered distributions are given by

$$\pi^{(\beta)}(x) = \frac{1}{Z(\beta)} \exp\left(\beta \sum_{x_i \sim x_j} x_i x_j + \mu \sum_i x_i\right). \tag{38}$$

This is an M^2 dimensional model which undergoes a phase transition as $M \to \infty$ at some critical temperature β_c. When $\mu = 0$ it is known that $\beta_c = \log(1 + \sqrt{2})/2$ [Baxter, 2007].

We observe that λ exhibits very different characteristics in this scenario compared to the Gaussian model: it is not monotonic and is maximized at the phase transition. Consequently, the optimal annealing schedule is denser near the phase transition. We also note from Figure 7 that both λ and Λ increases roughly linearly with respect to M. Given Proposition 1, this is to be expected even if this result is not directly applicable here as the target distribution does not factorize. As N increases, the round trip rate of reversible PT decays to 0 and non-reversible PT increase towards $\bar{\tau}$ as seen in Figure 8. This is consistent with Theorem 3.

7.3 Effects of ELE violation

As discussed in Section 3.3, we do not expect (A2) to hold. Increasing the number n_{expl} of MCMC exploration steps taken between two communication steps (see Algorithm 1) can be used to approach ELE. However a priori one may be concerned that n_{expl} would have to be very large to do so.

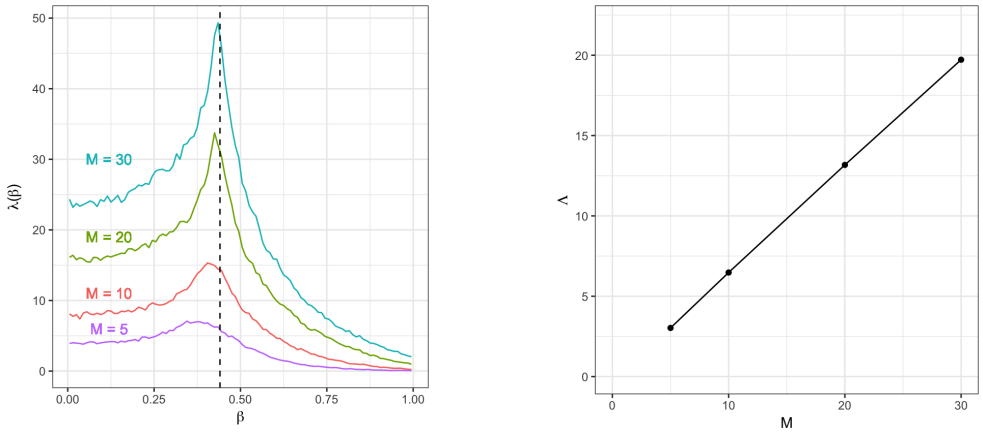

Figure 7: Estimate of the local communication barrier (left) and global communication barrier (right) for the Ising model with $\mu = 0$ and $M = 5, 10, 20, 30$. The vertical line is at the phase transition.

To investigate empirically whether our methodology is robust to the violation of the ELE assumption, we run the non-reversible method with different values for n_{expl}. Let d_{var} denote the number of variables in the model. We run experiments with $n_{\text{expl}} = 0$, $(1/2)d_{\text{var}}$, d_{var}, $2d_{\text{var}}$, $4d_{\text{var}}$, ..., $32d_{\text{var}}$. The fractions $0, 1/2, 1, 2, \ldots$ involved in this construction can be interpreted as the expected number of times an individual variable is updated in an exploration phase, i.e. the *expected updates per exploration phase*. The only exception is for the reference chain ($\beta = 0$), we always use $n_{\text{expl}} = 1$ since we can get exact samples from π_0 in our experiments. The case $n_{\text{expl}} = 0$ for $\beta > 0$ technically still yields an ergodic chain thanks to the communication steps.

We look at the Ising model under the effect of a magnetic field. We set the magnetic moment $\mu = 0.1$, leading to a target distribution where all marginals assign a mass of less than 0.07 to $x_i = 1$. The results are displayed in Figure 9. Even in this multi-modal problem, we observe a strong resilience of Algorithm 3 to violations of ELE.

We conjecture that this resilience may come from the structure of typical neighbourhoods of non-reversible PT. Our intuition can be described using a point process defined in a two-dimensional space, where one axis indexes PT communication iterations, and the other axis consists of the parallel chain indices. The set of points in the process encodes the rejected swaps that occurred in the entire execution of the PT algorithm. In the regime of a large number of parallel chains, for a given location in this point process, a neighbourhood will contain with high probability either zero or one rejection event. The key observation is that in both cases, no two chains interact more than once in the neighbourhood. This is true by direct inspection of Figure 10. As a consequence, even when a small number of exploration steps are used between swaps, with high probability they will accumulate by the time a pair of chains meet again (in a different neighbourhood), since each fixed neighbourhood will contain an increasing number of PT iterations in the limiting regime described in Section 6.

The same is not true for reversible PT, where the typical local neighbourhood can contain

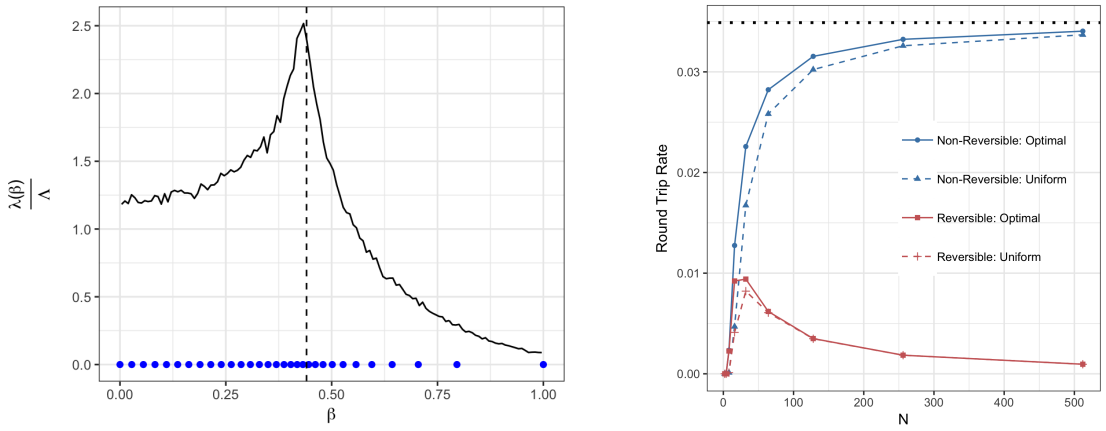

Figure 8: (Left) Optimal annealing schedule for the Ising model with $M = 20$, $\Lambda = 13.33$ with $N = 30$. The vertical line is at the phase transition. (Right) The round trip rates when $M = 20$ with a uniform schedule (dashed) to the optimal schedule (solid) for both non-reversible (blue) and reversible (red) PT. The dotted horizontal line represents the approximation of the optimal round trip rate $\hat{\tau}$.

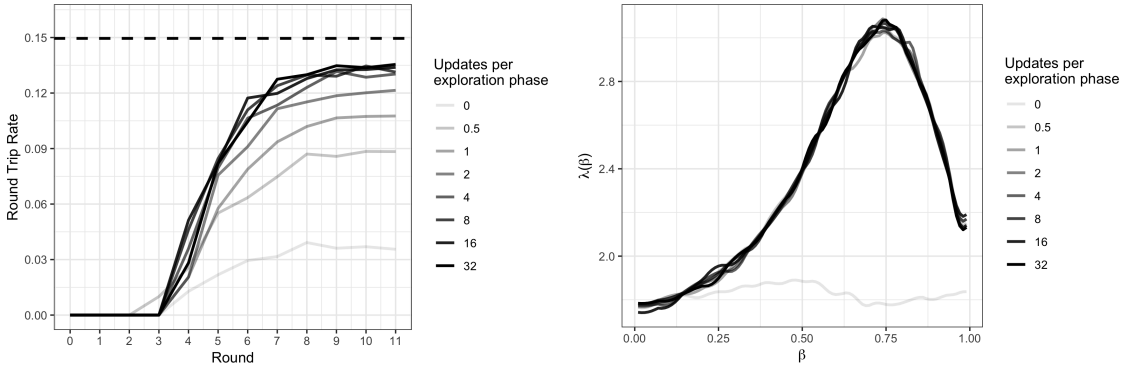

Figure 9: Ising model ($M = 5, \mu = 0.1$) with $N = 16$ and $\hat{\Lambda} = 2.35$. (Left) Round trip rate directly measured from the empirical replica trajectories and estimated $\bar{\tau}$, identified with the dotted line. (Right) Estimate of the local communication barrier $\lambda(\beta)$. Whenever $n_{\text{expl}} > 0$, the adaptive scheme accurately learns $\bar{\tau}, \lambda$.

an arbitrary large number of events, and hence pairs of chains can interact more than once in the neighbourhood. As a consequence, we conjecture that for our non-reversible results, it may be possible to significantly weaken the ELE assumption, but not for reversible PT. We leave the theoretical investigation of this question for future work.

To provide some empirical justification to this conjecture, we performed another experiment on the magnetic Ising model, fixing the expected updates per exploration phase to 1/2 and increasing the number of chains instead. The results are displayed in Figure 11 and support that by increasing the number of parallel chains, the actual round trip rate still converges to the theoretical bound from below even in the face of severe ELE violation.

Figure 10: Zoom on the bottom right panel of Figure 1 (columns are PT iterations, rows are parallel chains) exemplifying the structure of typical neighbourhoods of non-reversible PT for large N. There are either no rejection events (left), or one rejection event (right). In both cases, no two chains interact more than once (piece-wise linear lines).

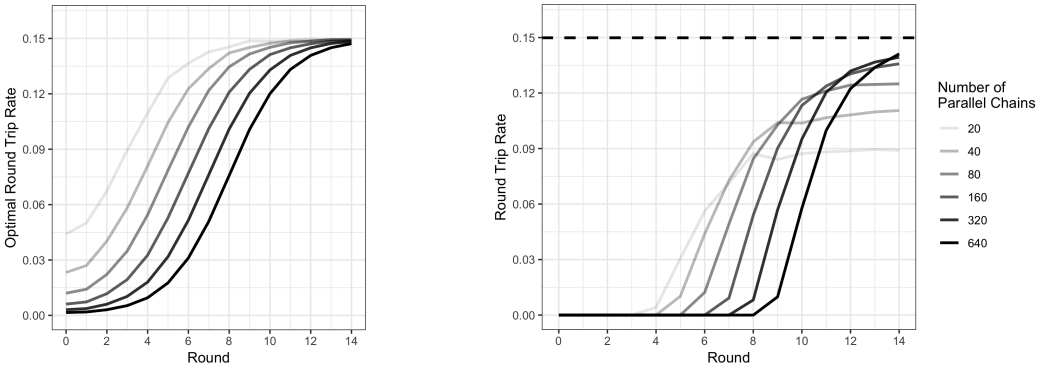

Figure 11: Impact of increasing the number N of parallel chains for an example where ELE is severely violated (only half the variables are updated at each exploration step). (Left) Estimated upper bound $\bar{\tau}$. (Right) Actual round trip rate directly measured from the empirical index process trajectories. The number of scans required for the round trip rates to stabilize increases with N as predicted by Proposition 1 but eventually reaches a higher round round trip.

7.4 Comparison with other parallel tempering schemes

In this section, we present results to support that the increased round trip rates enjoyed by our method does indeed translate into increased ESS per compute time. The following experiments benchmark the empirical running time of Algorithm 3 compared to previous adaptive PT methods [Atchadé et al., 2011, Miasojedow et al., 2013].

The methods we considered are: (1) the stochastic optimization adaptive method for reversible schemes proposed in [Atchadé et al., 2011]; (2), a second stochastic optimization scheme, which still selects the optimal number of chains using the 23% rule but uses an improved update scheme from [Miasojedow et al., 2013], refer to Appendix G.3 for details; (3) our adaptive non reversible PT scheme; and finally, (4) our scheme, combined with a better initialization based on a preliminary execution of a sequential Monte Carlo algorithm (more precisely, based on a "sequential change of measure," labelled SCM, as described in [Del Moral et al., 2006]), we use this to investigate the effect on the violation of the stationarity assumption, and for fairness, we use this sophisticated initialization method for all the methods except (3). We benchmarked the methods on four models: (a) a 369-dimensional hierarchical model applied to a dataset of rocket launch failure/success indicator variables [McDowell, 2019]; (b) a 19-dimensional Spike-and-Slab variable selection model applied

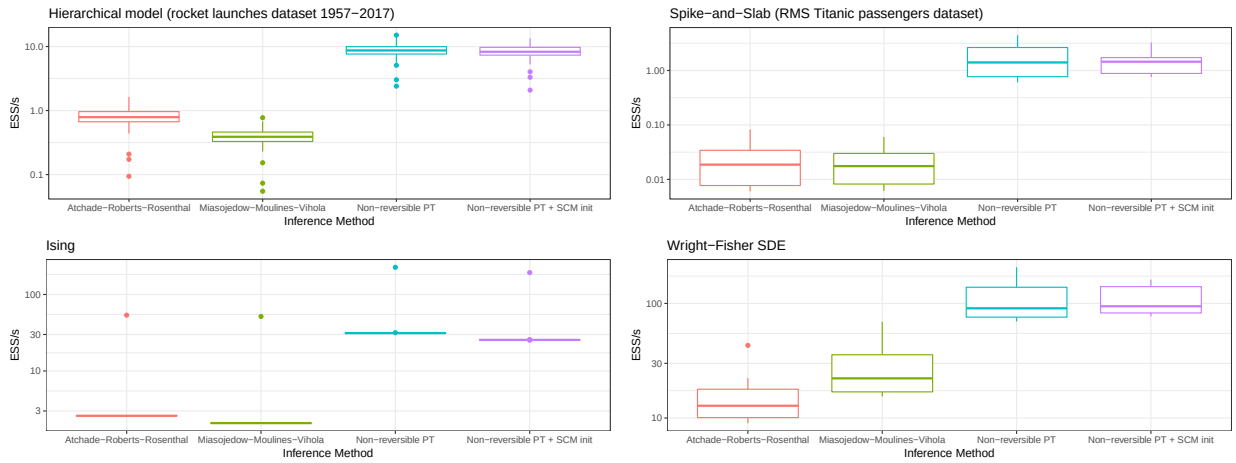

Figure 12: Effective Sample Size (ESS) per second (ordinate, in log scale) for four PT adaptive methods (abscissa). The four facets show results for the four models described in Section 7.4.

to the RMS Titanic Passenger Manifest dataset [Hind, 2019]; (c) A 25-dimensional Ising model from Section 7.2 ($M = 5$); (d) a 9-dimensional model for an end-point conditioned Wright-Fisher stochastic differential equation (see, e.g., [Tataru et al., 2017]).

We refer the reader to Appendix G for implementation details, experimental setup, and detailed description of the models and datasets used.

In Figure 12, each dot summarized in the box plots represents the ESS per total wall clock time in seconds including adaptation time for the marginal of one of the model variables. The results show that adaptation is more efficient with our proposed non-reversible scheme, with a speed-up in the 10–100 range for the four models considered. The results also suggest that SMC-based initialization may not have a large impact on the performance of our method.

8 References

[Altekar et al., 2004] Altekar, G., Dwarkadas, S., Huelsenbeck, J. P., and Ronquist, F. (2004). Parallel Metropolis coupled Markov chain Monte Carlo for Bayesian phylogenetic inference. *Bioinformatics*, 20(3):407–415.

[Andrec et al., 2005] Andrec, M., Felts, A. K., Gallicchio, E., and Levy, R. M. (2005). Protein folding pathways from replica exchange simulations and a kinetic network model. *Proceedings of the National Academy of Sciences*, 102(19):6801–6806.

[Atchadé et al., 2011] Atchadé, Y. F., Roberts, G. O., and Rosenthal, J. S. (2011). Towards optimal scaling of Metropolis-coupled Markov chain Monte Carlo. *Statistics and Computing*, 21(4):555–568.

[Ballnus et al., 2017] Ballnus, B., Hug, S., Hatz, K., Görlitz, L., Hasenauer, J., and Theis, F. J. (2017). Comprehensive benchmarking of Markov chain Monte Carlo methods for dynamical systems. *BMC Systems Biology*, 11(1):63.

[Baxter, 2007] Baxter, R. J. (2007). *Exactly Solved Models in Statistical Mechanics*. Dover books on Physics. Dover Publications.

[Bierkens et al., 2018] Bierkens, J., Bouchard-Côté, A., Doucet, A., Duncan, A. B., Fearnhead, P., Lienart, T., Roberts, G. O., and Vollmer, S. J. (2018). Piecewise deterministic Markov processes for scalable Monte Carlo on restricted domains. *Statistics & Probability Letters*, 136:148–154.

[Billingsley, 2013] Billingsley, P. (2013). *Convergence of Probability Measures*. John Wiley & Sons.

[Böttcher et al., 2013] Böttcher, B., Schilling, R., and Wang, J. (2013). Lévy matters. iii, volume 2099 of Lecture Notes in Mathematics.

[Chen et al., 1999] Chen, F., Lovász, L., and Pak, I. (1999). Lifting Markov chains to speed up mixing. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 275–281. ACM.

[Cheon and Liang, 2008] Cheon, S. and Liang, F. (2008). Phylogenetic tree construction using sequential stochastic approximation Monte Carlo. *BioSystems*, 91(1):94–107.

[Cho et al., 2010] Cho, K., Raiko, T., and Ilin, A. (2010). Parallel tempering is efficient for learning restricted Boltzmann machines. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8. IEEE.

[Davidson-Pilon, 2015] Davidson-Pilon, C. (2015). *Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference*. Addison-Wesley Professional, New York, 1st edition edition.

[Davis, 1993] Davis, M. H. (1993). *Markov Models & Optimization*. Chapman and Hall.

[Del Moral et al., 2006] Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 68(3):411–436.

[Desjardins et al., 2014] Desjardins, G., Luo, H., Courville, A., and Bengio, Y. (2014). Deep tempering. *arXiv preprint arXiv:1410.0123*.

[Diaconis et al., 2000] Diaconis, P., Holmes, S., and Neal, R. M. (2000). Analysis of a nonreversible Markov chain sampler. *The Annals of Applied Probability*, 10(3):726–752.

[Dupuis et al., 2012] Dupuis, P., Liu, Y., Plattner, N., and Doll, J. D. (2012). On the infinite swapping limit for parallel tempering. *SIAM Multiscale Modeling & Simulation*, 10(3):986–1022.

[Earl and Deem, 2005] Earl, D. J. and Deem, M. W. (2005). Parallel tempering: Theory, applications, and new perspectives. *Physical Chemistry Chemical Physics*, 7(23):3910–3916.

[Ethier and Kurtz, 2009] Ethier, S. N. and Kurtz, T. G. (2009). *Markov Processes: Characterization and Convergence*, volume 282. John Wiley & Sons.

[Fang et al., 2014] Fang, Y., Feng, S., Tam, K.-M., Yun, Z., Moreno, J., Ramanujam, J., and Jarrell, M. (2014). Parallel tempering simulation of the three-dimensional Edwards-Anderson model with compact asynchronous multispin coding on GPU. *Computer Physics Communications*, 185(10):2467–2478.

[Friesen, 2015] Friesen, J. (2015). *Java Threads and the Concurrency Utilities*. Apress, Berkeley, CA, USA, 1st edition.

[Fritsch and Carlson, 1980] Fritsch, F. and Carlson, R. (1980). Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246.

[Geyer, 1991] Geyer, C. J. (1991). Markov chain Monte Carlo maximum likelihood. *Interface Proceedings*.

[Hind, 2019] Hind, P. (2019). Rms titanic passenger dataset. data retrieved from https://tinyurl.com/y55c8kc7.

[Hukushima and Nemoto, 1996] Hukushima, K. and Nemoto, K. (1996). Exchange Monte Carlo method and application to spin glass simulations. *Journal of the Physical Society of Japan*, 65(6):1604–1608.

[Kallenberg, 2002] Kallenberg, O. (2002). *Foundations of Modern Probability*. Springer, 2nd edition.

[Katzgraber et al., 2006] Katzgraber, H. G., Trebst, S., Huse, D. A., and Troyer, M. (2006). Feedback-optimized parallel tempering Monte Carlo. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(03):P03018.

[Kofke, 2002] Kofke, D. A. (2002). On the acceptance probability of replica-exchange Monte Carlo trials. *The Journal of Chemical Physics*, 117(15):6911–6914.

[Kone and Kofke, 2005] Kone, A. and Kofke, D. A. (2005). Selection of temperature intervals for parallel-tempering simulations. *The Journal of Chemical Physics*, 122(20):206101.

[Lacki and Miasojedow, 2016] Lacki, M. K. and Miasojedow, B. (2016). State-dependent swap strategies and automatic reduction of number of temperatures in adaptive parallel tempering algorithm. *Statistics and Computing*, 26(5):951–964.

[Leiserson et al., 2012] Leiserson, C. E., Schardl, T. B., and Sukha, J. (2012). Deterministic parallel random-number generation for dynamic-multithreading platforms. *MIT web domain*.

[Lelièvre et al., 2010] Lelièvre, T., Stoltz, G., and Rousset, M. (2010). *Free Energy Computations: A Mathematical Perspective*. World Scientific.

[Lingenheil et al., 2009] Lingenheil, M., Denschlag, R., Mathias, G., and Tavan, P. (2009). Efficiency of exchange schemes in replica exchange. *Chemical Physics Letters*, 478(1-3):80–84.

[McDowell, 2019] McDowell, J. (2019). Launch logs. data retrieved from https://tinyurl.com/y5veq9yf.

[Miasojedow et al., 2013] Miasojedow, B., Moulines, E., and Vihola, M. (2013). An adaptive parallel tempering algorithm. *Journal of Computational and Graphical Statistics*, 22(3):649–664.

[Mingas and Bouganis, 2012] Mingas, G. and Bouganis, C.-S. (2012). Parallel tempering MCMC acceleration using reconfigurable hardware. In Choy, O. C. S., Cheung, R. C. C., Athanas, P., and Sano, K., editors, *Reconfigurable Computing: Architectures, Tools and Applications*, Lecture Notes in Computer Science, pages 227–238. Springer Berlin Heidelberg.

[Nadler and Hansmann, 2007] Nadler, W. and Hansmann, U. H. E. (2007). Dynamics and optimal number of replicas in parallel tempering simulations. *Physical Review E*, 76(6):065701.

[Neal, 2003] Neal, R. M. (2003). Slice sampling. *The Annals of Statistics*, 31(3):705–767.

[Okabe et al., 2001] Okabe, T., Kawata, M., Okamoto, Y., and Mikami, M. (2001). Replica-exchange Monte Carlo method for the isobaric–isothermal ensemble. *Chemical Physics Letters*, 335(5-6):435–439.

[Pitera and Swope, 2003] Pitera, J. W. and Swope, W. (2003). Understanding folding and design: Replica-exchange simulations of"trp-cage"miniproteins. *Proceedings of the National Academy of Sciences*, 100(13):7587–7592.

[Predescu et al., 2004] Predescu, C., Predescu, M., and Ciobanu, C. V. (2004). The incomplete beta function law for parallel tempering sampling of classical canonical systems. *The Journal of Chemical Physics*, 120(9):4119–4128.

[Roberts and Rosenthal, 2014] Roberts, G. O. and Rosenthal, J. S. (2014). Minimising MCMC variance via diffusion limits, with an application to simulated tempering. *The Annals of Applied Probability*, 24(1):131–149.

[Sakai and Hukushima, 2016] Sakai, Y. and Hukushima, K. (2016). Irreversible simulated tempering. *Journal of the Physical Society of Japan*, 85(10):104002.

[Saranen and Seikkala, 1988] Saranen, J. and Seikkala, S. (1988). Solution of a nonlinear two-point boundary value problem with Neumann-type boundary data. *Journal of Mathematical Analysis and Applications*, 135(2):691–701.

[Steele and Lea, 2013] Steele, G. and Lea, D. (2013). Splittable random application programming interface. https://docs.oracle.com/javase/8/docs/api/java/util/SplittableRandom.html. [Online; accessed 6-May-2019].

[Swendsen and Wang, 1986] Swendsen, R. H. and Wang, J.-S. (1986). Replica Monte Carlo simulation of spin-glasses. *Physical Review Letters*, 57(21):2607.

[Tataru et al., 2017] Tataru, P., Simonsen, M., Bataillon, T., and Hobolth, A. (2017). Statistical inference in the Wright-Fisher model using allele frequency data. *Systematic Biology*, 66(1):e30–e46.

[Turitsyn et al., 2011] Turitsyn, K. S., Chertkov, M., and Vucelja, M. (2011). Irreversible Monte Carlo algorithms for efficient sampling. *Physica D: Nonlinear Phenomena*, 240(4-5):410–414.

[Vucelja, 2016] Vucelja, M. (2016). Lifting: a nonreversible Markov chain Monte Carlo algorithm. *American Journal of Physics*, 84(12):958–968.

[Wu, 2017] Wu, F. (2017). Irreversible Parallel Tempering and an Application to a Bayesian Nonparametric Latent Feature Model. Master's thesis, Oxford University.

[Zilberstein, 1996] Zilberstein, S. (1996). Using Anytime Algorithms in Intelligent Systems. *AI Magazine*, 17(3):73–73.

Appendix A Invariant distribution of PT chain and index process

We show here that the Markov chain $(\mathbf{X}_n, \sigma_n)$ on the augmented state space $\mathcal{X}^{N+1} \times \mathrm{Perm}(N)$ is invariant with respect to the distribution,

$$\bar{\boldsymbol{\pi}}(\mathbf{x}, \sigma) = \frac{1}{(N+1)!}\boldsymbol{\pi}(\mathbf{x}). \tag{39}$$

To establish this result, we reformulate the swap kernel in the augmented space as

$$\mathbf{K}^{(i,j)}(\bar{\mathbf{x}}, A \times \{\sigma'\}) = \left(1 - \alpha^{(i,j)}(\mathbf{x})\right) \delta_{\mathbf{x}}(A)\mathbb{I}[\sigma' = \sigma] + \alpha^{(i,j)}(\mathbf{x})\delta_{\mathbf{x}^{(i,j)}}(A)\mathbb{I}[\sigma' = (i\ j) \circ \sigma], \tag{40}$$

where $\bar{\mathbf{x}} = (\mathbf{x}, \sigma)$ denotes an augmented state, $(i\ j) \in \mathrm{Perm}(N)$ denotes a transposition (swap) between i and j, and $(i\ j) \circ \sigma$ is the composition of σ followed by the swap $(i\ j)$. We slightly abuse notation here and denote the kernel in the augmented space with the same symbol. The exploration kernel does not cause swaps, so in the permutation-augmented space we set it to

$$\mathbf{K}^{\mathrm{expl}}((\mathbf{x}, \sigma), A_0 \times A_1 \times \ldots A_N \times \{\sigma'\}) = \mathbb{I}[\sigma' = \sigma]\mathbf{K}^{\mathrm{expl}}(\mathbf{x}, A_0 \times A_1 \times \ldots A_N), \tag{41}$$

with a similar abuse of notation.

Since $\mathbf{K}_n^{\mathrm{PT}} = \mathbf{K}_n^{\mathrm{comm}}\mathbf{K}^{\mathrm{expl}}$, it is enough to verify that both $\mathbf{K}^{\mathrm{expl}}$ and $\mathbf{K}_n^{\mathrm{comm}}$ are $\bar{\boldsymbol{\pi}}$-invariant to show $\mathbf{K}_n^{\mathrm{PT}}$ is $\bar{\boldsymbol{\pi}}$-invariant. It is clear by construction that $\mathbf{K}^{\mathrm{expl}}$ defined by (3) is $\bar{\boldsymbol{\pi}}$-stationary. It remains to verify that $\mathbf{K}_n^{\mathrm{comm}}$ is also $\bar{\boldsymbol{\pi}}$-stationary. Clearly $\mathbf{K}^{\mathrm{SEO}}, \mathbf{K}_n^{\mathrm{DEO}}$ are trivially $\bar{\boldsymbol{\pi}}$-invariant if each swap kernel $\mathbf{K}^{(i,j)}$ is. We verify this directly. Let $\bar{\mathbf{x}} = (\mathbf{x}, \sigma) \in \mathcal{X}^{N+1} \times \mathrm{Perm}([N])$, then

$$\begin{aligned}
&\int_{\mathcal{X}^{N+1}\times\mathrm{Perm}([N])} \bar{\boldsymbol{\pi}}(\mathrm{d}\bar{\mathbf{x}})\mathbf{K}^{(i,j)}(\bar{\mathbf{x}}, A \times \{\sigma'\}) && (42)\\
&\quad= \frac{1}{(N+1)!}\sum_{\sigma}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\mathbf{K}^{(i,j)}(\bar{\mathbf{x}}, A \times \{\sigma'\}) && (43)\\
&\quad= \frac{1}{(N+1)!}\sum_{\sigma}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\left(1 - \alpha^{(i,j)}(\mathbf{x})\right)\delta_{\mathbf{x}}(A)\mathbb{I}[\sigma' = \sigma]\\
&\qquad+ \frac{1}{(N+1)!}\sum_{\sigma}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\alpha^{(i,j)}(\mathbf{x})\delta_{\mathbf{x}^{(i,j)}}(A)\mathbb{I}[\sigma' = (i,j) \circ \sigma] && (44)\\
&\quad= \frac{1}{(N+1)!}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\sum_{\sigma}\left(1 - \alpha^{(i,j)}(\mathbf{x})\right)\delta_{\mathbf{x}}(A)\mathbb{I}[\sigma' = \sigma]\\
&\qquad+ \frac{1}{(N+1)!}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\sum_{\sigma}\alpha^{(i,j)}(\mathbf{x})\delta_{\mathbf{x}^{(i,j)}}(A)\mathbb{I}[\sigma' = (i,j) \circ \sigma] && (45)\\
&\quad= \frac{1}{(N+1)!}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\left\{\left(1 - \alpha^{(i,j)}(\mathbf{x})\right)\delta_{\mathbf{x}}(A) + \alpha^{(i,j)}(\mathbf{x})\delta_{\mathbf{x}^{(i,j)}}(A)\right\} && (46)\\
&\quad= \frac{1}{(N+1)!}\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\mathbf{K}^{(i,j)}(\mathbf{x}, A) && (47)
\end{aligned}$$

By construction, $\int_{\mathcal{X}^{N+1}} \boldsymbol{\pi}(\mathrm{d}\mathbf{x})\mathbf{K}^{(i,j)}(\mathbf{x}, A) = \boldsymbol{\pi}(A)$, therefore

$$\int_{\mathcal{X}^{N+1}\times\mathrm{Perm}([N])} \bar{\boldsymbol{\pi}}(\mathrm{d}\bar{\mathbf{x}})\mathbf{K}^{(i,j)}(\bar{\mathbf{x}}, A\times\{\sigma'\}) = \frac{1}{(N+1)!}\boldsymbol{\pi}(A) = \bar{\boldsymbol{\pi}}(A\times\{\sigma'\}) \tag{48}$$

and thus $\mathbf{K}^{(i,j)}$ is $\bar{\boldsymbol{\pi}}$-invariant.

Appendix B Markov kernel for the index process

For SEO, initialize $Y_0 = (I_0, \varepsilon_0)$ where $I_0 = i$ and $\varepsilon_0 \sim \mathrm{Unif}\{-1, 1\}$. Define the Markov transition kernel, $Y_{n+1}|Y_n \sim P^{\mathrm{SEO}}(Y_n, \cdot)$ in two steps. First simulate

$$I_{n+1}|Y_n = (i,\varepsilon) \sim \begin{cases} (i+\varepsilon)\wedge N \vee 0 & \text{with probability } s^{(i,i+\varepsilon)}, \\ i & \text{otherwise,} \end{cases} \tag{49}$$

where the expression "$\wedge N \vee 0$" enforces the annealing parameter boundaries. In the second step, independently sample $\varepsilon_{n+1} \sim \mathrm{Unif}\{-1, +1\}$.

Similarly for DEO, initialize $I_0 = i$ and $\varepsilon_0 = 1$ if i is even and -1 otherwise. Analogous to the SEO construction, we define $Y_{n+1}|Y_n \sim P^{\mathrm{DEO}}(Y_n, \cdot)$ in two steps. We first update $I_{n+1}|Y_n = (i,\varepsilon)$ as in (49), but apply the deterministic update in the second step,

$$\varepsilon_{n+1} = \begin{cases} \varepsilon & \text{if } I_{n+1} = i+\varepsilon, \\ -\varepsilon & \text{otherwise.} \end{cases} \tag{50}$$

Appendix C Proof of Theorem 1

To simplify notation for the rest of the proof, let $T_\uparrow$ and $T_\downarrow$ be the hitting times to the target and reference defined by,

$$T_\uparrow = \min\{n : (I_n, \varepsilon_n) = (N, 1)\}, \quad T_\downarrow = \min\{n : (I_n, \varepsilon_n) = (0, -1)\}. \tag{51}$$

We will also denote

$$s_i = s^{(i-1,i)}, \qquad r_i = r^{(i-1,i)}. \tag{52}$$

(a) If we define $a_\bullet^i = \mathbb{E}_{\mathrm{SEO}}(T_\bullet | I_0 = i)$ for $i = 0, \ldots, N$ and $\bullet \in \{\uparrow, \downarrow\}$, then we have

$$\mathbb{E}_{\mathrm{SEO}}(T) = a_\uparrow^0 + a_\downarrow^N. \tag{53}$$

By the Markov property, for $i = 1, \ldots, N-1$, $a^i_\bullet$ satisfies the recursion

$$a^i_\bullet = \frac{1}{2} s_{i+1}(a^{i+1}_\bullet + 1) + \frac{1}{2} s_i (a^{i-1}_\bullet + 1) + \frac{1}{2}(r_{i+1} + r_i)(a^i_\bullet + 1). \tag{54}$$

For $i = 1, \ldots, N$, we substitute in $b^i_\bullet = a^i_\bullet - a^{i-1}_\bullet$ into (54). After simplification, $b^i_\bullet$ satisfies the following recursive relation

$$-2 = s_{i+1} b^{i+1}_\bullet - s_i b^i_\bullet. \tag{55}$$

The solutions to (55) are

$$s_i b^i_\bullet = s_1 b^1_\bullet - 2(i-1), \tag{56}$$

or equivalently

$$s_i b^i_\bullet = s_N b^N_\bullet + 2(N-i). \tag{57}$$

We now deal with the case of $\uparrow$ and $\downarrow$ separately.

(a) To determine $a^0_\uparrow$, we note that if $I_0 = 0$ then $I_1 = 1$ with probability $\frac{1}{2}s_1$ and $I_1 = 0$ otherwise. So $a^0_\uparrow$ satisfies

$$a^0_\uparrow = \frac{1}{2} s_1 (a^1_\uparrow + 1) + \left(1 - \frac{1}{2} s_1\right)(a^0_\uparrow + 1), \tag{58}$$

or equivalently

$$s_1 b^1_\uparrow = -2. \tag{59}$$

Substituting this into (56) implies $s_i b^i_\uparrow = -2i$. By summing $b^i_\uparrow = a^i_\uparrow - a^{i-1}_\uparrow$ from $i = 1, \ldots, N$ and, noting $a^N_\uparrow = 0$, we get

$$a^0_\uparrow = \sum_{i=1}^{N} \frac{2i}{s_i}. \tag{60}$$

(b) Similarly to determine $a^N_\downarrow$ we note that if $I_0 = N$ then $I_1 = N-1$ with probability $\frac{1}{2}s_N$ and $I_1 = N$ otherwise. So $a^N_\downarrow$ satisfies

$$a^N_\downarrow = \frac{1}{2} s_N (a^{N-1}_\downarrow + 1) + \left(1 - \frac{1}{2} s_N\right)(a^N_\downarrow + 1), \tag{61}$$

or equivalently

$$s_N b_\downarrow^N = 2. \tag{62}$$

Substituting this into (57) implies $s_i b_\downarrow^i = 2 + 2(N-i)$. By summing $b_\downarrow^i = a_\downarrow^i - a_\downarrow^{i-1}$ from $i = 1, \ldots, N$ and, noting $a_\downarrow^0 = 0$, we get

$$a_\downarrow^N = \sum_{i=1}^{N} \frac{2(N-i)+2}{s_i}. \tag{63}$$

Substituting in (60) and (63) into (53), it follows that

$$\mathbb{E}_{\mathrm{SEO}}(T) = \sum_{i=1}^{N} \frac{2i}{s_i} + \sum_{i=1}^{N} \frac{2(N-i)+2}{s_i} \tag{64}$$

$$= 2(N+1) \sum_{i=1}^{N} \frac{1}{s_i} \tag{65}$$

$$= 2N(N+1) + 2(N+1) \sum_{i=1}^{N} \frac{r_i}{s_i}. \tag{66}$$

(b) If we define $a_\bullet^{i,\varepsilon} = \mathbb{E}_{\mathrm{DEO}}(T_\bullet | I_0 = i, \varepsilon_0 = \varepsilon)$ for $i = 0, \ldots, N$, $\varepsilon \in \{+,-\}$ and $\bullet \in \{\uparrow, \downarrow\}$, then we have

$$\mathbb{E}_{\mathrm{DEO}}(T) = a_\uparrow^{0,-} + a_\downarrow^{N,+}. \tag{67}$$

Note that for $i = 1, \ldots, N-1$ $a_\bullet^{i,\varepsilon}$ satisfies the recursion relations

$$a_\bullet^{i,+} = s_{i+1}(a_\bullet^{i+1,+} + 1) + r_{i+1}(a_\bullet^{i,-} + 1), \tag{68}$$

$$a_\bullet^{i,-} = s_i(a_\bullet^{i-1,-} + 1) + r_i(a_\bullet^{i,-} + 1). \tag{69}$$

If we substitute $c_\bullet^i = a_\bullet^{i,+} + a_\bullet^{i-1,-}$, and $d_\bullet^i = a_\bullet^{i,+} - a_\bullet^{i-1,-}$ into (68) and (69) and simplify, we obtain

$$a_\bullet^{i+1,+} - a_\bullet^{i,+} = r_{i+1} d_\bullet^{i+1} - 1, \tag{70}$$

$$a_\bullet^{i,-} - a_\bullet^{i-1,-} = r_i d_\bullet^i + 1. \tag{71}$$

By subtracting and adding (70) and (71), we obtain a joint recursion relation for $c_\bullet^i$ and $d_\bullet^i$ of the form

$$c_\bullet^{i+1} - c_\bullet^i = r_{i+1} d_\bullet^{i+1} + r_i d_\bullet^i, \tag{72}$$

$$d_\bullet^{i+1} - d_\bullet^i = r_{i+1} d_\bullet^{i+1} + r_i d_\bullet^i - 2. \tag{73}$$

Note that (73) can be rewritten as

$$s_{i+1}d_\bullet^{i+1} - s_i d_\bullet^i = -2. \tag{74}$$

Once one has expressions for $c_\bullet^i$ and $d_\bullet^i$, then we can recover $a_\bullet^{i,\varepsilon}$ by using

$$a_\bullet^{i,+} = \frac{c_\bullet^i + d_\bullet^i}{2}, \tag{75}$$

$$a_\bullet^{i-1,-} = \frac{c_\bullet^i - d_\bullet^i}{2}. \tag{76}$$

We now deal with the $\uparrow$ and $\downarrow$ cases separately.

(a) Note that $a_\uparrow^{0,-} = a_\uparrow^{0,+} + 1$. We can substitute this into (70) to get $s_1 d_\uparrow^1 = -2$, which combined with (74) implies

$$s_i d_\uparrow^i = -2i. \tag{77}$$

Since $a_\uparrow^{N,+} = 0$ we have $c_\uparrow^N = -d_\uparrow^N$, so by summing (72) we get

$$2a_\uparrow^{0,-} = c_\uparrow^1 - d_\uparrow^1 \tag{78}$$

$$= c_\uparrow^N - d_\uparrow^1 - \sum_{i=1}^{N-1}(c_\uparrow^{i+1} - c_\uparrow^i) \tag{79}$$

$$= -d_\uparrow^N - d_\uparrow^1 - \sum_{i=1}^{N-1}(r_{i+1}d_\uparrow^{i+1} + r_i d_\uparrow^i) \tag{80}$$

$$= -s_N d_\uparrow^N - s_1 d_\uparrow^1 - 2\sum_{i=1}^{N} r_i d_\uparrow^i. \tag{81}$$

After substituting in (77) into (81), we obtain

$$a_\uparrow^{0,-} = N + 1 + \sum_{i=1}^{N} \frac{2ir_i}{s_i}. \tag{82}$$

(b) Note that $a_\downarrow^{N,+} = a_\uparrow^{N,-} + 1$. We can substitute this expression into (71) to get $s_N d_\downarrow^N = 2$, which combined with (74) implies

$$s_i d_\downarrow^i = 2(N - i + 1). \tag{83}$$

Since $a_{\downarrow}^{0,-} = 0$ we have $c_{\downarrow}^{1} = d_{\downarrow}^{1}$, so by summing (72) we get

$$2a_{\downarrow}^{N,+} = c_{\downarrow}^{N} + d_{\downarrow}^{N} \tag{84}$$

$$= c_{\downarrow}^{1} + d_{\downarrow}^{N} + \sum_{i=1}^{N-1} (c_{\downarrow}^{i+1} - c_{\downarrow}^{i}) \tag{85}$$

$$= d_{\downarrow}^{1} + d_{\downarrow}^{N} + \sum_{i=1}^{N-1} (r_{i+1} d_{\downarrow}^{i+1} + r_i d_{\downarrow}^{i}) \tag{86}$$

$$= s_1 d_{\downarrow}^{1} + s_N d_{\downarrow}^{N} + 2 \sum_{i=1}^{N} r_i d_{\downarrow}^{i}. \tag{87}$$

After substituting in (83) into (87), we obtain

$$a_{\downarrow}^{N,+} = N + 1 + \sum_{i=1}^{N} \frac{2(N-i+1) r_i}{s_i}. \tag{88}$$

Finally, by substituting (82) and (88) into (67), it follows that

$$\mathbb{E}_{\mathrm{DEO}}(T) = 2(N+1) + 2(N+1) \sum_{i=1}^{N} \frac{r_i}{s_i}. \tag{89}$$

We begin by recalling the following estimate for $r(\beta, \beta')$ from Proposition 1 in [Predescu et al., 2004]:

Proposition 2. *[Predescu et al., 2004] Suppose V^3 is integrable with respect to π_0 and π. For $\beta \leq \beta'$, let $\bar{\beta} = \frac{\beta+\beta'}{2}$, then we have*

$$r(\beta, \beta') = (\beta' - \beta)\lambda(\bar{\beta}) + O(|\beta' - \beta|^3), \tag{90}$$

where λ satisfies (25).

When $\beta < \beta'$, $(\beta' - \beta)\lambda(\bar{\beta})$ is the Riemann sum for $\int_{\beta}^{\beta'} \lambda(b) db$ with a single rectangle. Let $C^k([0,1])$ be the set of k times continuously differentiable function on $[0,1]$. If $\lambda \in C^2([0,1])$, then standard midpoint rule error estimates yield

$$\left| \int_{\beta}^{\beta'} \lambda(b) \mathrm{d}b - (\beta' - \beta)\lambda(\bar{\beta}) \right| \leq \frac{1}{12} \left\| \frac{\mathrm{d}^2 \lambda}{\mathrm{d}\beta^2} \right\|_{\infty} |\beta' - \beta|^3. \tag{91}$$

Therefore, if $\lambda \in C^2([0,1])$ then we can substitute (91) in (90) in Proposition 2, to obtain Theorem 2. This follows from Proposition 3.

Proposition 3. *If V^k is integrable with respect to π_0 and π, then $\lambda \in C^{k-1}([0,1])$.*

Proof. Suppose V^k is integrable with respect to π_0 and π, we want to show here that $\lambda : [0,1] \to \mathbb{R}_+$ given by

$$\lambda(\beta) = \frac{1}{2}\int_{\mathcal{X}^2} |V(x) - V(y)|\pi^{(\beta)}(x)\pi^{(\beta)}(y)\mathrm{d}x\mathrm{d}y \tag{92}$$

is in $C^{k-1}([0,1])$. If we define $L(x,y) = L(x)L(y)$ and $\pi_0(x,y) = \pi_0(x)\pi_0(y)$, we can rewrite (92) as

$$\lambda(\beta) = \frac{1}{2\mathcal{Z}(\beta)^2}\int_{\mathcal{X}^2} |V(x) - V(y)|L(x,y)^\beta\pi_0(x,y)\mathrm{d}x\mathrm{d}y \tag{93}$$

$$= \frac{g(\beta)}{2\mathcal{Z}(\beta)^2}, \tag{94}$$

where $\mathcal{Z}, g : [0,1] \to \mathbb{R}_+$ are defined by

$$\mathcal{Z}(\beta) = \int_{\mathcal{X}} L(x)^\beta\pi_0(x)\mathrm{d}x, \tag{95}$$

$$g(\beta) = \int_{\mathcal{X}^2} |V(x) - V(y)|L(x,y)^\beta\pi_0(x,y)\mathrm{d}x\mathrm{d}y. \tag{96}$$

Since $\mathcal{Z}(\beta) > 0$ on $[0,1]$, if we can show that $\mathcal{Z}, g \in C^{k-1}([0,1])$ then it implies that $\lambda \in C^{k-1}([0,1])$. This is established in Lemma 2. □

Lemma 1. *If V^k is integrable with respect to π_0 and π for $k \in \mathbb{N}$. Then for all $\beta \in [0,1]$, $j \leq k$, V^j is $\pi^{(\beta)}$-integrable.*

Proof. We begin by noting that for $L > 0$ and $\beta \in [0,1]$, we have $L^\beta \leq 1 + L$. This implies

$$\int_{\mathcal{X}} |V(x)|^k\pi^{(\beta)}(x)\mathrm{d}x \tag{97}$$

$$= \frac{1}{\mathcal{Z}(\beta)}\int_{\mathcal{X}} |V(x)|^k L(x)^\beta\pi_0(x)\mathrm{d}x \tag{98}$$

$$\leq \frac{1}{\mathcal{Z}(\beta)}\int_{\mathcal{X}} |V(x)|^k\pi_0(x)\mathrm{d}x + \frac{1}{\mathcal{Z}(\beta)}\int_{\mathcal{X}} |V(x)|^k L(x)\pi_0(x)\mathrm{d}x \tag{99}$$

$$= \frac{\mathcal{Z}(0)}{\mathcal{Z}(\beta)}\int_{\mathcal{X}} |V(x)|^k\pi_0(x)\mathrm{d}x + \frac{\mathcal{Z}(1)}{\mathcal{Z}(\beta)}\int_{\mathcal{X}} |V(x)|^k\pi(x)\mathrm{d}x \tag{100}$$

$$< \infty. \tag{101}$$

Therefore since V^k is π_0 and π-integrable, V^k is $\pi^{(\beta)}$-integrable. Finally by Jensen's inequality we have for $j \geq k$,

$$\int_{\mathcal{X}} |V(x)|^j\pi^{(\beta)}(x)\mathrm{d}x \leq \left(\int_{\mathcal{X}} |V(x)|^k\pi^{(\beta)}(x)\mathrm{d}x\right)^{\frac{j}{k}} < \infty. \tag{102}$$

□

Lemma 2. *Suppose V^k is integrable with respect to π_0 and π for some $k \in \mathbb{N}$ then:*

(a) $\mathcal{Z} \in C^k([0,1])$ with derivatives satisfying,

$$\frac{d^j \mathcal{Z}}{d\beta^j} = \int_{\mathcal{X}} (-1)^j V(x)^j L(x)^\beta \pi_0(x) \mathrm{d}x, \tag{103}$$

for $j \leq k$.

(b) $g \in C^{k-1}([0,1])$ with derivatives satisfying,

$$\frac{d^j g}{d\beta^j} = \int_{\mathcal{X}^2} (-1)^j |V(x) - V(y)|(V(x) + V(y))^j L(x,y)^\beta \pi_0(x,y) \mathrm{d}x\mathrm{d}y, \tag{104}$$

for $j < k$.

Proof. (a) Let $h(x,\beta) = L(x)^\beta \pi_0(x) = \exp(-\beta V(x))\pi_0(x)$ which satisfies

$$\frac{\partial^j}{\partial \beta^j} h(x,\beta) = (-1)^j V(x)^j L(x)^\beta \pi_0(x). \tag{105}$$

Note for all $\beta \in [0,1]$ and $j \leq k$,

$$\sup_{\beta \in [0,1]} \left| \frac{\partial^j}{\partial \beta^j} h(x,\beta) \right| \leq |V(x)|^j \pi_0(x) + |V(x)|^j L(x) \pi_0(x). \tag{106}$$

The left hand side of (106) dominates $\frac{\partial^j h}{\partial \beta^j}$ uniformly in β and is integrable by Lemma 1. The result follows using the Leibniz integration rule.

(b) Let $\tilde{h}(x,y,\beta) = |V(x) - V(y)| L(x,y)^\beta \pi_0(x,y)$. By using $\log L(x,y) = -V(x) - V(y)$, we get

$$\frac{\partial^j}{\partial \beta^j} \tilde{h}(x,y,\beta) = (-1)^j |V(x) - V(y)|(V(x) + V(y))^j L(x,y)^\beta \pi_0(x,y). \tag{107}$$

Similar to (a), we have for all $\beta \in [0,1]$, $j \leq k-1$,

$$\begin{aligned} \sup_{\beta \in [0,1]} \left| \frac{\partial^j}{\partial \beta^j} \tilde{h}(x,y,\beta) \right| \leq\ & |V(x) - V(y)||V(x) + V(y)|^j \pi_0(x,y) \\ & + |V(x) - V(y)||V(x) + V(y)|^j L(x,y) \pi_0(x,y), \end{aligned} \tag{108}$$

The left hand side of (108) dominates $\frac{\partial^j \tilde{h}}{\partial \beta^j}$ uniformly in β. It is integrable by Lemma 1 and using the fact that V^k is integrable with respect to π_0 and π. The result follows using the Leibniz integration rule.

□

Appendix D Proof of Proposition 1

For $k = 1, 2$, let us define $\mathbf{V}_k^{(\beta)} \stackrel{d}{=} \mathbf{V}(X_k^{(\beta)})$ where $X_k^{(\beta)} \sim \pi_d^{(\beta)}$ and $\mathbf{V}(x) = \sum_{i=1}^d V(x_i)$. The independence structure from Equation (28) tells us that $\mathbf{V}_k^{(\beta)}$ can be decomposed as $\mathbf{V}_k^{(\beta)} = \sum_{i=1}^d V_{ki}^{(\beta)}$ where $V_{ki}^{(\beta)}$ are iid with a distribution identical to $V^{(\beta)}$, and therefore

$$\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)} = \sum_{i=1}^d V_{1i}^{(\beta)} - V_{2i}^{(\beta)}. \tag{109}$$

The random variables $\{V_{1i}^{(\beta)} - V_{2i}^{(\beta)}\}_{i=1}^d$ are iid with mean zero and variance $2\sigma^2(\beta)$. By the central limit theorem,

$$\frac{\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}}{\sqrt{2\sigma^2(\beta)d}} = \frac{1}{\sqrt{d}} \sum_{i=1}^d \frac{V_{1i}^{(\beta)} - V_{2i}^{(\beta)}}{\sqrt{2\sigma^2(\beta)}} \stackrel{}{\Longrightarrow}_{d\to\infty} \tilde{Z} \sim N(0,1). \tag{110}$$

Thus we have

$$\lambda_d(\beta) = \frac{1}{2}\mathbb{E}\left[|\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}|\right] \tag{111}$$

$$= \frac{1}{2}\sqrt{2\sigma^2(\beta)d}\,\mathbb{E}\left[\left|\frac{\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}}{\sqrt{2\sigma^2(\beta)d}}\right|\right]. \tag{112}$$

The sequence of variables indexed by d in the expectation in (112) is also uniformly integrable. This follows by noting that the second moment of the integrand in (112) is uniformly bounded in d:

$$\sup_d \mathbb{E}\left[\left|\frac{\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}}{\sqrt{2\sigma^2(\beta)d}}\right|^2\right] = \sup_d \frac{1}{2\sigma^2(\beta)d} \sum_{i=1}^d \mathrm{Var}\left[V_{1i}^{(\beta)} - V_{2i}^{(\beta)}\right] = 1. \tag{113}$$

By $d \to \infty$ and using (110) we have

$$\lim_{d\to\infty} \sqrt{\frac{2}{\sigma^2(\beta)d}}\lambda_d(\beta) = \mathbb{E}|\tilde{Z}| = \sqrt{\frac{2}{\pi}}, \tag{114}$$

which proves (29).

To obtain the dimensional scaling limit for Λ_d, we use Cauchy-Schwarz

$$\frac{\lambda_d(\beta)}{\sqrt{d}} = \frac{1}{2\sqrt{d}}\mathbb{E}\left[|\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}|\right] \tag{115}$$

$$\leq \frac{1}{2\sqrt{d}}\sqrt{\mathrm{Var}\left[|\mathbf{V}_1^{(\beta)} - \mathbf{V}_2^{(\beta)}|\right]} \tag{116}$$

$$= \frac{\sigma(\beta)}{\sqrt{2}}. \tag{117}$$

Finally, (29), (117) along with dominated convergence yield

$$\lim_{d\to\infty}\frac{\Lambda_d}{\sqrt{d}}=\int_0^1\lim_{d\to\infty}\frac{\lambda_d(\beta)}{\sqrt{d}}\mathrm{d}\beta=\int_0^1\frac{\sigma(\beta)}{\sqrt{\pi}}\mathrm{d}\beta. \tag{118}$$

Appendix E Proof of Theorem 3

We first note that (b) and (c) follow immediately from (a) and Corollary 1. So to prove Theorem 3 it is sufficient to show (a).

Let $\mathcal{P}_N=\{\beta_0,\ldots,\beta_N\}$. There exists an i_0 such that $\mathcal{P}_{N+1}=\mathcal{P}_N\cup\{\beta\}$ for some $\beta_{i_0}<\beta<\beta_{i_0+1}$. Therefore, we have

$$E(\mathcal{P}_{N+1})-E(\mathcal{P}_N)=\frac{r(\beta_{i_0},\beta)}{s(\beta_{i_0},\beta)}+\frac{r(\beta,\beta_{i_0+1})}{s(\beta,\beta_{i_0+1})}-\frac{r(\beta_{i_0},\beta_{i_0+1})}{s(\beta_{i_0},\beta_{i_0+1})} \tag{119}$$

$$=\frac{r(\beta_{i_0},\beta)s(\beta,\beta_{i_0+1})+s(\beta_{i_0},\beta)r(\beta,\beta_{i_0+1})}{s(\beta_{i_0},\beta)s(\beta,\beta_{i_0+1})}-\frac{r(\beta_{i_0},\beta_{i_0+1})}{s(\beta_{i_0},\beta_{i_0+1})} \tag{120}$$

$$\le\frac{r(\beta_{i_0},\beta)+r(\beta,\beta_{i_0+1})}{s(\beta_{i_0},\beta)s(\beta,\beta_{i_0+1})}-\frac{r(\beta_{i_0},\beta_{i_0+1})}{s(\beta_{i_0},\beta_{i_0+1})} \tag{121}$$

$$\le\frac{r(\beta_{i_0},\beta)+r(\beta,\beta_{i_0+1})}{1-r(\beta_{i_0},\beta)-r(\beta,\beta_{i_0+1})}-\frac{r(\beta_{i_0},\beta_{i_0+1})}{s(\beta_{i_0},\beta_{i_0+1})}. \tag{122}$$

The last inequality holds since

$$s(\beta_{i_0},\beta)s(\beta,\beta_{i_0+1})=(1-r(\beta_{i_0},\beta))(1-r(\beta,\beta_{i_0+1})) \tag{123}$$

$$\ge 1-r(\beta_{i_0},\beta)-r(\beta,\beta_{i_0+1}). \tag{124}$$

By Theorem 2 we have

$$r(\beta_{i_0},\beta)+r(\beta,\beta_{i_0+1})=r(\beta_{i_0},\beta_{i_0+1})+O(\|\mathcal{P}_N\|^3). \tag{125}$$

This implies

$$E(\mathcal{P}_{N+1})-E(\mathcal{P}_N)\le\frac{r(\beta_{i_0},\beta_{i_0+1})+O(\|\mathcal{P}_N\|^3)}{s(\beta_{i_0},\beta_{i_0+1})+O(\|\mathcal{P}_N\|^3)}-\frac{r(\beta_{i_0},\beta_{i_0+1})}{s(\beta_{i_0},\beta_{i_0+1})}. \tag{126}$$

As $\|\mathcal{P}_N\|\to 0$, the right hand side is asymptotically equivalent to zero. Therefore $E(\mathcal{P}_{N+1})\lesssim E(\mathcal{P}_N)$ as $\|\mathcal{P}_N\|\to 0$ and $E(\mathcal{P}_N)$ is asymptotically decreasing.

To show that $E(\mathcal{P}_N)$ asymptotically decreases to Λ, we use the fact that for all $\mathcal{P}_N$

$$\sum_{i=1}^N r^{(i-1,i)}\le E(\mathcal{P}_N)\le\frac{1}{\min_j s^{(j-1,j)}}\sum_{i=1}^N r^{(i-1,i)}. \tag{127}$$

By Theorem 2, we have $\min_j s_j=1+O(\|\mathcal{P}_N\|)$ and, by Corollary, we have 2 $\sum_{i=1}^N r^{(i-1,i)}=$

$\Lambda + O(N\|\mathcal{P}_N\|^3)$ which combined with (127) implies

$$E(\mathcal{P}_N) = \Lambda + O(\|\mathcal{P}_N\|). \tag{128}$$

Therefore $E(\mathcal{P}_N)$ converges to Λ at a $O(\|\mathcal{P}_N\|)$ rate as $\|\mathcal{P}_N\| \to 0$.

Appendix F Scaling limit of index process

F.1 Scaled index process

To establish scaling limits for $Y_n = (I_n, \varepsilon_n)$, it will be convenient to work in a continuous time setting. To do this, we suppose the times that PT iterations occur are distributed according to a Poisson process $\{M(\cdot)\}$ with mean μ_N. The number of PT iterations that occur by time $t \geq 0$ satisfies $M(t) \sim \text{Poisson}(\mu_N t)$. We define the *scaled index process* by $Z^N(t) = (W^N(t), \varepsilon^N(t))$ where $W^N(t) = I_{M(t)}/N$ and $\varepsilon^N(t) = \varepsilon_{M(t)}$.

For convenience, we will denote $\beta_w = G(w)$ and use $z = (w, \varepsilon) \in [0,1] \times \{-1, 1\}$ to be a *scaled index*. Define $C(\mathbb{R}_+, \mathcal{S})$ and $D(\mathbb{R}_+, \mathcal{S})$ to be set of functions $f : \mathbb{R}_+ \to \mathcal{S}$ that are continuous and càdlàg respectively.

The process $Z^N \in D(\mathbb{R}_+, [0,1] \times \{-1,1\})$ takes values on the discrete set $\mathcal{P}_{\text{uniform}} \times \{-1,1\}$ and is only well-defined when $Z^N(0) = z_0 \in \mathcal{P}_{\text{uniform}} \times \{-1,1\}$. To establish convergence, it is useful to extend it to a process Z^N which can be initialized at any $z_0 \in [0,1] \times \{-1,1\}$. Suppose $Z^N(0) = z_0 \in [0,1] \times \{-1,1\}$ and let $T_1, T_2, \ldots$ be the iteration times generated by the Poisson process M. We construct $Z^N(t)$ as follows: define $Z^N(t) = z_n$ for $t \in [T_n, T_{n+1})$ and update $z_{n+1}|z_n$ via a transition kernel dependent on the communication scheme. We determine this transition kernel mirroring the construction from Section 3.5.

Before doing this, it will be useful to define the backward and forward shift operators $\Phi_-^N, \Phi_+^N : [0,1] \to [0,1]$ by,

$$\Phi_-^N(w) = \begin{cases} w - \frac{1}{N} & w \in \left[\frac{1}{N}, 1\right], \\ \frac{1}{N} - w & w \in \left[0, \frac{1}{N}\right), \end{cases} \tag{129}$$

and similarly,

$$\Phi_+^N(w) = \begin{cases} w + \frac{1}{N} & w \in \left[0, 1 - \frac{1}{N}\right], \\ 1 - \left(\frac{1}{N} - (1-w)\right) & w \in \left(1 - \frac{1}{N}, 1\right]. \end{cases} \tag{130}$$

Intuitively $\Phi_\varepsilon^N(w)$ represents the location in $[0,1]$ after w moves a distance $\frac{1}{N}$ in the direction of ε with a reflection at 0 and 1.

F.2 Scaled index process for reversible PT

Under the SEO communication scheme, if $z_n = (w_n, \varepsilon_n) \in \{0, 1/N, \ldots, 1\} \times \{-1, 1\}$, then we have $w_{n+1} = \Phi^N_{\varepsilon_n}(w_n)$ if a swap successfully occurred and $w_{n+1} = w_n$ otherwise. In both cases, $\varepsilon_{n+1} \sim \text{Unif}\{-1, +1\}$. Since $\Phi^N_\varepsilon(w)$ is not only well-defined for $w \in \mathcal{P}_{\text{uniform}}$ but for $w \in [0, 1]$, we naturally extend this construction to any $w \in [0, 1]$.

Formally, we generate $(w_{n+1}, \varepsilon_{n+1})$ in two steps. In the first step we simulate,

$$w_{n+1}|w_n, \varepsilon_n \sim \begin{cases} \Phi^N_{\varepsilon_n}(w_n) & \text{with probability } s(\beta_{w_n}, \beta_{\Phi^N_{\varepsilon_n}(w_n)}), \\ w_n & \text{otherwise.} \end{cases} \tag{131}$$

In the second step we simulate $\varepsilon_{n+1} \sim \text{Unif}\{-1, +1\}$. This defines a continuous time Markov pure jump process $W^N \in D(\mathbb{R}_+, [0, 1])$ with jumps occurring according to an exponential of rate μ_N and is well defined when initialized at any state $w_0 \in [0, 1]$.

From Theorem 19.2 in [Kallenberg, 2002], the infinitesimal generator for W^N with SEO communication is

$$\mathcal{L}_{W^N} f(w) = \frac{\mu_N}{2} \sum_{\varepsilon \in \{\pm 1\}} \big(f(\Phi^N_\varepsilon(w)) - f(w)\big)\, s(\beta_w, \beta_{\Phi^N_\varepsilon(w)}), \tag{132}$$

where the domain $\mathcal{D}(\mathcal{L}_{W^N})$ is given by the set of functions such that $\mathcal{L}_{W^N} f$ is continuous. Since Φ^N_+, Φ^N_- are continuous, we have $\mathcal{D}(\mathcal{L}_{W^N}) = C([0, 1])$.

F.3 Scaled index process for non-reversible PT

Before defining the transition kernel for the scaled index process under DEO communication, it will be convenient to define the propagation function $\Phi^N : [0, 1] \times \{-1, 1\} \to [0, 1] \times \{-1, 1\}$ for $z = (w, \varepsilon)$,

$$\Phi^N(z) = \begin{cases} (\Phi^N_\varepsilon(w), \varepsilon) & \text{if } \Phi^N_\varepsilon(w) = w + \frac{\varepsilon}{N}, \\ (\Phi^N_\varepsilon(w), -\varepsilon) & \text{otherwise,} \end{cases} \tag{133}$$

and similarly the rejection function $R : [0, 1] \times \{-1, 1\} \to [0, 1] \times \{-1, 1\}$,

$$R(z) = (w, -\varepsilon). \tag{134}$$

Under the DEO scheme, if $z_n = (w_n, \varepsilon_n) \in \mathcal{P}_{\text{uniform}} \times \{-1, 1\}$, then we have $z_{n+1} = \Phi^N(z_n)$ when a swap is accepted and $z_{n+1} = R(z_n)$ otherwise. Since $\Phi^N(z)$ and $R(z)$ are well-defined for all of $z \in [0, 1] \times \{-1, 1\}$, we naturally extend this construction to any $z \in [0, 1] \times \{-1, 1\}$.

Formally, we generate z_{n+1} according to the transition kernel,

$$z_{n+1}|z_n \sim \begin{cases} \Phi^N(z_n) & \text{with probability } s(\beta_{w_n}, \beta_{\Phi^N_{\varepsilon_n}(w_n)}), \\ R(z_n) & \text{otherwise.} \end{cases} \tag{135}$$

This defines a continuous time Markov pure jump process $Z^N \in D(\mathbb{R}_+, [0,1] \times \{-1,1\})$ with jumps occurring at an exponential of rate μ_N. This process is well defined when initialized at any $z_0 \in [0,1] \times \{-1,1\}$.

Analogously to the reversible case, under DEO communication, the infinitesimal generator for Z^N is

$$\mathcal{L}_{Z^N} f(z) = \mu_N \left(f(\Phi^N(z)) - f(z) \right) s(\beta_w, \beta_{\Phi^N_\varepsilon(w)}) + \mu_N \left(f(R(z)) - f(z) \right) r(\beta_w, \beta_{\Phi^N_\varepsilon(w)}), \tag{136}$$

where $z = (w, \varepsilon)$ and $\mathcal{D}(\mathcal{L}_{Z^N})$ is given by the set of functions f such that $\mathcal{L}_{Z^N} f$ is continuous. Since Φ^N has discontinuities at $(\frac{1}{N}, -1)$ and $(1 - \frac{1}{N}, 1)$, we can verify that $f \in \mathcal{D}(\mathcal{L}_{Z^N})$ if and only if $f(w_0, -1) = f(w_0, 1)$ for $w_0 \in \{0, 1\}$.

F.4 Weak limits for scaled index processes

Define $W \in C(\mathbb{R}_+, [0,1])$ to be the diffusion on $[0,1]$ with generator

$$\mathcal{L}_W f(w) = \frac{1}{2} \frac{\mathrm{d}^2 f}{\mathrm{d} w^2}, \tag{137}$$

where the domain $\mathcal{D}(\mathcal{L}_W)$ is the set of functions $f \in C^2([0,1])$ such that $f'(0) = f'(1) = 0$. W is a Brownian motion on $[0,1]$ with reflective boundary conditions admitting the uniform distribution $\mathrm{Unif}([0,1])$ as stationary distribution.

Define $Z \in C(\mathbb{R}_+, [0,1] \times \{-1,1\})$ to be the PDMP on $[0,1] \times \{-1,1\}$ given by $Z(t) = (W(t), \varepsilon(t))$ where $W(t)$ moves in $[0,1]$ with velocity $\varepsilon(t)$ and the sign of $\varepsilon(t)$ is reversed at the arrivals times of a non-homogeneous Poisson process of rate $\lambda(G(W(t)))G'(W(t))$ or when $W(t)$ reaches the boundary $\{(0,-1), (1,+1)\}$; see [Bierkens et al., 2018] for a discussion of PDMP on restricted domains. The infinitesimal generator of Z is given by

$$\mathcal{L}_Z f(z) = \varepsilon \frac{\partial f}{\partial w}(z) + \lambda(\beta_w) G'(w) \left(f(R(z)) - f(z) \right), \tag{138}$$

for any $f \in \mathcal{D}(\mathcal{L}_Z)$, the set of functions $f \in C^1([0,1] \times \{-1,1\})$ such that $f(w_0, -1) = f(w_0, 1)$ and $\frac{\partial f}{\partial w}(w_0, -1) = -\frac{\partial f}{\partial w}(w_0, 1)$ for $w_0 \in \{0, 1\}$.

F.5 Proof of scaling limit for reversible PT

We will prove Theorem 4(a) by using Theorem 17.25 from [Kallenberg, 2002].

Theorem 5 (Trotter, Sova, Kurtz, Mackevičius)**.** *Let $X, X^1, X^2, \dots$ be Feller processes defined on a state space S with generators $\mathcal{L}, \mathcal{L}_1, \mathcal{L}_2, \dots$ respectively. If D is a core for $\mathcal{L}$, then the following statements are equivalent:*

1. *If $f \in D$, there exist $f_N \in \mathcal{D}(\mathcal{L}_N)$ such that $\|f_N - f\|_\infty \to 0$ and $\|\mathcal{L}_N f_N - \mathcal{L} f\|_\infty \to 0$ as $N \to \infty$.*
2. *If $X^N(0)$ converges weakly to $X(0)$ in S, then X^N converges weakly to X in $D(\mathbb{R}_+, S)$.*

We will be applying Theorem 5 with $\mathcal{L} = \mathcal{L}_W$ defined as $\mathcal{L}_W f = \frac{1}{2} f''$ for $f \in \mathcal{D}(\mathcal{L}_W)$ where

$$\mathcal{D}(\mathcal{L}_W) := \left\{ f \in C^2([0,1]) : f'(0) = f'(1) = 0 \right\}, \tag{139}$$

and $\mathcal{L}_N = \mathcal{L}_{W^N}$ defined in (132), which we recall here for the reader's sake

$$\mathcal{L}_{W^N} f(w) = \frac{N^2}{2} \sum_{\varepsilon \in \{\pm 1\}} \left(f(\Phi_\varepsilon^N(w)) - f(w) \right) s(\beta_w, \beta_{\Phi_\varepsilon^N(w)}), \quad w \in [0,1], \tag{140}$$

with $\Phi_\pm^N(w)$ defined in (129), (130) and $\beta_w = G(w)$. Recall from the discussion just before (132) that $\mathcal{L}_{W^N}$ defines a Feller semigroup.

First notice that in [Kallenberg, 2002], the transition semi-group and generator of a Feller process taking values in a metric space S are defined on $C_0(S)$, the space of functions vanishing at infinity. Equivalently $f \in C_0(S)$ if and only for any $\delta > 0$ there exists a compact set $K \subset S$ such that for $x \notin K$, $|f(x)| < \delta$. In our case since $S = [0,1]$ is compact $C_0(S) = C(S)$, which justifies the definition of the generator $\mathcal{L}_W$ given above.

F.5.1 The Feller property of $\mathcal{L}_W$.

Similarly $\mathcal{L}_W$ can be seen to define a Feller semigroup on $C([0,1])$ by the Hille-Yosida theorem; see [Kallenberg, 2002, Theorem 19.11].

Indeed the first condition is satisfied since any function $f \in C([0,1])$ can be uniformly approximated within $\epsilon > 0$ by a polynomial p_ϵ, that is a smooth function, by the Stone-Weirstrass theorem. We can further uniformly approximate p_ϵ within ϵ by a C^2 function $\hat{p}_\epsilon$ with vanishing derivatives at the endpoints. For example one can let, for a δ to be chosen later, $\hat{p}_\epsilon(x) = p_\epsilon(x)$ for $x \in (\delta, 1-\delta)$ and for $x \le \delta$ set $\hat{p}_\epsilon(x) = \int_0^x \rho_\delta(y) p_\epsilon'(y) \mathrm{d}y + c$, where ρ_δ is a smooth, increasing transition function such that $\rho_\delta(x) = 0$ for $x < 0$, $\rho_\delta(x) = 1$ for $x > \delta$, for example let $\rho_\delta = \rho(x/\delta)$, $\rho(x) = g(x)/(g(x) + g(1-x))$ and $g(x) = \exp(-1/x)\mathbf{1}_{\{x>0\}}$. We chose c so that $\hat{p}_\epsilon(x)$ is continuous at δ. A similar construction can be used for the right-endpoint. One can then check that indeed $\hat{p}_\epsilon \in C^2([0,1])$, $\hat{p}_\epsilon'(0) = \hat{p}_\epsilon'(1) = 0$ and that for δ small enough $\|\hat{p}_\epsilon - p_\epsilon\|_\infty < \epsilon$.

The second condition of [Kallenberg, 2002, Theorem 19.11] requires that for some $\mu > 0$, the set $(\mu - \mathcal{L}_W)(\mathcal{D}(\mathcal{L}_W))$ is dense in $C([0,1])$. Let $g \in C([0,1])$ be given. We apply [Saranen and Seikkala, 1988, Corollary 2.2], with $f(t,y) = 2\mu y - 2g$, which is clearly square integrable in t and

2μ-Lipschitz in y. Then [Saranen and Seikkala, 1988, Corollary 2.2] implies that for small enough $\mu > 0$ the two-point Neumann-boundary value problem

$$\mu u - \frac{1}{2}u'' = g$$
$$u'(0) = u'(1) = 0$$

admits a solution in the Sobolev space $H^2([0,1])$ of functions with square integrable first and second derivatives. This already implies that $u \in C^1([0,1])$, whereas the continuity of g and of u a priori implies the continuity of u'' since $u'' = 2\mu u - g$. Overall, for any $g \in C([0,1])$ we can find $u \in \mathcal{D}(\mathcal{L}_W)$ such that $g = (\mu - \mathcal{L}_W)g$ establishing the second condition of [Kallenberg, 2002, Theorem 19.11].

The third condition of [Kallenberg, 2002, Theorem 19.11] is that $(\mathcal{L}_W, \mathcal{D}(\mathcal{L}_W))$ satisfies the *positive maximum principle*, that is if for some $f \in \mathcal{D}(\mathcal{L}_W))$ and $x_0 \in [0,1]$ we have $f(x_0) \geq f(x) \vee 0$ for all $x \in [0,1]$, then $f''(x_0) \leq 0$. Suppose first that the maximum is attained at an interior point $x_0 \in (0,1)$; since $f \in C^2([0,1])$, by definition of $\mathcal{D}(\mathcal{L}_W))$, $f''(x_0) \geq 0$. If on the other hand the positive maximum is attained at $x_0 = 0$, suppose that $f''(0) > \epsilon$ for all $x \leq \epsilon$. Thus for $0 < y < \epsilon$ small enough, since $f'(0) = 0$ we have

$$f(y) = f(0) + \int_0^y f'(s)\mathrm{d}s = f(0) + \int_0^y \int_0^s f''(r)\mathrm{d}r\mathrm{d}y \geq f(0) + \frac{\epsilon}{2}y^2 > f(0),$$

thus arriving at a contradiction.

We have thus established that $(\mathcal{L}_W, \mathcal{D}(\mathcal{L}_W))$ satisfies all conditions of [Kallenberg, 2002, Theorem 19.11] and therefore generates a Feller process. Now we can apply Theorem 5 to prove Theorem 4(a). We only need to check the first condition of Theorem 5. In this direction, first note that by definition $\Phi^N_\pm(w) = w \pm 1/N$ for $w \in [1/N, 1 - 1/N]$. Thus in this case using a Taylor expansion we have for $w^*_- \in [w - 1/N, w]$ and $w^*_+ \in [w, w + 1/N]$ that

$$\begin{aligned} f(\Phi^N_+(w)) - 2f(w) + f(\Phi^N_-(w)) &= f(w) + \frac{1}{N}f'(w) + \frac{1}{2N^2}f''(w^*_+) \\ &\quad + f(w) - \frac{1}{N}f'(w) + \frac{1}{2N^2}f''(w^*_-) - 2f(w) \qquad (141) \\ &= \frac{1}{2N^2}\left(f''(w^*_+) + f''(w^*_-)\right). \qquad (142) \end{aligned}$$

Since f'' is uniformly continuous it follows that as $N \to \infty$,

$$\sup_{w \in [0,1]} |f''(w^*_\pm) - f''(w)| = o(1), \qquad (143)$$

and therefore for $w \in [1/N, 1 - 1/N]$ we have

$$\sup_{w \in [0,1]} \left| f(\Phi^N_+(w)) - 2f(w) + f(\Phi^N_-(w)) - \frac{f''(w)}{N^2} \right| = o\left(\frac{1}{N^2}\right). \qquad (144)$$

When $w \in [0, 1/N)$ or $w \in (1 - 1/N, 1]$ we instead perform a Taylor expansion around 0 or 1 respectively. We only do the calculation in the first case, the other case being similar. Let $w \in [0, 1/N)$ in which case, since $f'(0) = 0$, for $w^*, w^*_-, w^*_+ \in [0, 2/N]$

$$\begin{aligned} f(\Phi^N_+(w)) - 2f(w) + f(\Phi^N_-(w)) &= f(0) + \Phi^N_+(w)f'(0) + \frac{1}{2}\left[\Phi^N_+(w)\right]^2 f''(w^*_+) \\ &\quad + f(0) + \Phi^N_-(w)f'(0) + \frac{1}{2}\left[\Phi^N_-(w)\right]^2 f''(w^*_-) \\ &\quad - 2f(0) - 2f'(0)w - 2\frac{f''(w^*)}{2}w^2 \qquad (145) \\ &= \frac{f''(0)}{2}\left\{\left[\Phi^N_+(w)\right]^2 + \left[\Phi^N_-(w)\right]^2 - 2w^2\right\} + o\left(N^{-2}\right), \qquad (146) \end{aligned}$$

where the error term is uniform in w and was obtained by combining the facts that f'' is uniformly continuous and that $|\Phi^N_\pm|, |w|/ \leq 2/N$. Finally notice that since $w \in [0, 1/N]$, then

$$\left[\Phi^N_+(w)\right]^2 + \left[\Phi^N_-(w)\right]^2 - 2w^2 = \left[w + \frac{1}{N}\right]^2 + \left[\frac{1}{N} - w\right]^2 - 2w^2 = \frac{2}{N^2}. \qquad (147)$$

Finally we will need the following weaker version of Theorem 2, whose proof is postponed to the end of this section.

Lemma 3. *Suppose that $\pi_0(|V|), \pi(|V|) < \infty$. Then there exists a constant $C > 0$ such that*

$$\sup_\beta |s(\beta, \beta + \delta) - 1| \leq C\delta. \qquad (148)$$

Using Lemma 3, we see that for some constant $C > 0$

$$\sup_{w\in[0,1]} \left|s\left(\beta_w, \beta_{\Phi^N_\pm(w)}\right) - 1\right| \leq C \sup_w \left|G(w) - G\left(\Phi^N_\pm(w)\right)\right| \leq \frac{C\|G'\|_\infty}{N}, \qquad (149)$$

and therefore

$$\begin{aligned} \mathcal{L}_N f(w) &= \frac{N^2}{2} \sum_{\varepsilon\in\{\pm 1\}} \left(f(\Phi^N_\varepsilon(w)) - f(w)\right) s(\beta_w, \beta_{\Phi^N_\varepsilon(w)}) \qquad (150) \\ &= \frac{N^2}{2} \sum_{\varepsilon\in\{\pm 1\}} \left(f(\Phi^N_\varepsilon(w)) - f(w)\right) \left[1 + o(N^{-1})\right] \qquad (151) \\ &= \frac{N^2}{2} \left(f(\Phi^N_+(w)) - 2f(w) + f(\Phi^N_-(w))\right) \left[1 + o(N^{-1})\right] = \frac{N^2}{2}\frac{f''(w)}{N^2}\left[1 + o(1)\right], \qquad (152) \end{aligned}$$

where the error term is uniform in w. Thus $\mathcal{L}_N f \to \mathcal{L} f$ uniformly.

of Lemma 3. Using the bound $0 \leq 1 - \exp(-x) \leq x$ for $x \geq 0$ we have

$$
\begin{aligned}
&\big|s(\beta,\beta')-1\big| \\
&\leq \int\int \pi^{(\beta)}(\mathrm{d}x)\pi^{(\beta')}(\mathrm{d}y)\Big[1-\exp\big(-\max\big\{0,(\beta'-\beta)[V(x)-V(y)]\big\}\big)\Big] && (153)\\
&\leq |\beta'-\beta|\int\int \pi^{(\beta)}(\mathrm{d}x)\pi^{(\beta')}(\mathrm{d}y)\max\big\{0,[V(x)-V(y)]\big\}\big)\Big] && (154)\\
&\leq |\beta'-\beta|\int\int \pi^{(\beta)}(\mathrm{d}x)\pi^{(\beta')}(\mathrm{d}y)\big(|V(x)|+|V(y)|\big)\Big] && (155)\\
&\leq 2|\beta'-\beta|\sup_{\beta}\pi^{(\beta)}(|V|). && (156)
\end{aligned}
$$

□

F.6 Proof of scaling limit for non-reversible PT

We will prove Theorem 4(b) in a slightly round about way. We will define the auxiliary processes $\{U^N(\cdot)\}$, $\{U(\cdot)\}$ living on the unit circle $\mathbb{S}^1 := \{z \in \mathbb{C} : |z| = 1\}$ along with a mapping $\phi : \mathbb{S}^1 \mapsto [0,1]\times\{\pm 1\}$ such that $Z^N = \phi(U^N)$ and $Z = \phi(U)$. We will first show that the law of U^N converges weakly to U.

Before defining the processes we point out that we will identify $\mathbb{S}^1$ with $[0, 2\pi)$ in the usual way by working in mod 2π arithmetic. Notice that in this way

$$
C(\mathbb{S}^1) = \{f \in C([0,2\pi]) : f(0) = f(2\pi)\}. \tag{157}
$$

The reason for working with these auxiliary processes is that we can now avoid working with PDMPs with boundaries, helping us to remove a layer of technicalities.

For any N we define $\Sigma^N : \mathbb{S}^1 \mapsto \mathbb{S}^1$ through $\Sigma^N(\theta) = \theta + 2\pi/N$. Consider then a continuous-time process U^N that jumps at the arrival times of a homogeneous Poisson process with rate N according to the kernel

$$
Q^N(\theta, \mathrm{d}\theta') = s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right)\delta_{\Sigma^N(\theta)}(\mathrm{d}\theta') + \left[1 - s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right)\right]\delta_{2\pi-\theta}(\mathrm{d}\theta'), \tag{158}
$$

where

$$
\tilde{\beta}_\theta = \begin{cases} G\left(\frac{\theta}{\pi}\right), & \theta \in [0,\pi), \\ G\left(\frac{2\pi-\theta}{\pi}\right), & \theta \in [\pi, 2\pi). \end{cases} \tag{159}
$$

Define the map

$$
\phi(\theta) = \begin{cases} \left(\frac{\theta}{\pi}, +1\right), & \theta \in [0,\pi), \\ \left(\frac{2\pi-\theta}{\pi}, -1\right), & \theta \in [\pi, 2\pi). \end{cases} \tag{160}
$$

Essentially we think of the circle as comprising of two copies of $[0,1]$ glued together at the end points. The top one is traversed in an increasing direction and the bottom one in a decreasing

direction. When glued together and viewed as a circle these dynamics translate in a counter-clockwise rotation with occasional reflections w.r.t. the x-axis at the time of events. With this picture in mind it should be clear that $\phi(U^N) = Z^N$.

We also define the limiting process U as follows. First let

$$\tilde{\lambda}(\theta) = (\lambda \circ G)(\phi^1(\theta))G'(\phi^1(\theta)), \tag{161}$$

where $\phi^1(\theta)$ is the first coordinate of $\phi(\theta)$. Notice at this point that $\phi^1 : \mathbb{S}^1 \mapsto [0,1]$ is continuous and satisfies $\phi^1(\theta) = \phi^1(-\theta)$ for any $\theta \in [0, 2\pi)$, whence we obtain that $\tilde{\lambda}(-\theta) = \tilde{\lambda}(\theta)$. Given $U(0) = \theta$, let T_1 be a random variable such that

$$\mathbb{P}[T_1 \geq t] = \exp\left\{-\int_0^t \tilde{\lambda}(\theta + s)\mathrm{d}s\right\}, \tag{162}$$

and define the process as $U(s) = \theta + s \mod 2\pi$ for all $s < T_1$ and set $U(T_1) = -U(T_1-) \mod 2\pi$. Iterating this procedure will define the $\mathbb{S}^1$-valued PDMP $\{U(\cdot)\}$. We first need the next lemma.

Lemma 4. *Suppose V is integrable with respect to π_0 and π. The process U defined above is a Feller process, its infinitesimal generator is given by*

$$\mathcal{L}_U f(\theta) = f'(\theta) + \tilde{\lambda}(\theta)\left[f(2\pi - \theta) - f(\theta)\right], \tag{163}$$

with domain

$$\mathcal{D}(\mathcal{L}_U) = \{f \in C^1([0,\pi]) : f(0) = f(2\pi)\}, \tag{164}$$

and invariant measure $\mathrm{d}\theta/2\pi$.

Proof. First, note that since $\mathbb{S}^1$ is compact $C_0(\mathbb{S}^1) = C(\mathbb{S}^1)$ and thus to study the Feller process we consider the semi-group $\{P_U^t\}_t$ defined by the process U as acting on $C(\mathbb{S}^1)$. To prove the Feller property we can thus use [Davis, 1993, Theorem 27.6]. Since there is no boundary in the definition of U the first assumption is automatically verified, $Qf(\theta) = f(-\theta) \in C\left(\mathbb{S}^1\right)$ for any continuous f. We also know that the rate $\tilde{\lambda}$ is bounded whereas by Proposition 3 and the fact that $G \in C^1[0,1]$ we know that $\tilde{\lambda}$ is also continuous. Therefore the third condition of [Davis, 1993, Theorem 27.6] holds and thus U is Feller.

The infinitesimal generator will be defined on $\mathcal{D}(\mathcal{L}_U) \subseteq C(\mathbb{S}^1)$. The domain is defined as the class of functions $f \in C(\mathbb{S}^1)$ such that

$$g(\theta) = \lim_{h \to 0} \frac{1}{h}\left[P_U^t f(\theta) - f(\theta)\right] \in C(\mathbb{S}^1), \tag{165}$$

where the limit is uniform in θ. However by [Böttcher et al., 2013, Theorem 1.33], we can also consider pointwise limits without enlarging the domain. Using the definition of U we then have for

$\theta \in [0, 2\pi)$ that

$$\frac{1}{h}\mathbb{E}^\theta \left[f(U_h) - f(\theta)\right] = \frac{1}{h}\left[f\left(\theta + h\right)\mathbb{P}^\theta \left[T_1 \geq h\right] - f(\theta)\right] + \frac{1}{h}\mathbb{E}^\theta \left[f\left(U_h\right)\mathbb{1}\left\{T_1 \leq h\right\}\right]. \tag{166}$$

Since for $x \geq 0$ we have $|\exp(-x) - 1 + x| \leq Cx^2$ for some constant $C > 0$, and using the uniform continuity of $\tilde{\lambda}$ we can see that

$$\left|\exp\left\{-\int_0^h \tilde{\lambda}(\theta + s)\mathrm{d}s\right\} - 1 + \tilde{\lambda}(\theta)h\right| = o(h), \tag{167}$$

uniformly in θ, and thus

$$\begin{aligned}\frac{1}{h}f\left(\theta + h\right)\mathbb{P}^\theta \left[T_1 \geq h\right] - f(\theta) &= \frac{1}{h}\left[f\left(\theta + h\right)\left[1 - \tilde{\lambda}(\theta)h + o(h)\right] - f(\theta)\right] &(168)\\ &= \frac{1}{h}\left[f(\theta + h) - f(\theta)\right] - \tilde{\lambda}(\theta)f(\theta) + o(1). &(169)\end{aligned}$$

In addition

$$\begin{aligned}&\frac{1}{h}\mathbb{E}^\theta \left[f\left(U_h\right)\mathbb{1}\left\{T_1 \leq h\right\}\right]\\ &= \frac{1}{h}\int_0^h \tilde{\lambda}(\theta + s)\exp\left\{-\int_0^s \tilde{\lambda}(\theta + r)\mathrm{d}r\right\}\mathrm{d}s P_U^{h-s}Qf(\theta) &(170)\\ &\to \tilde{\lambda}(\theta)Qf(\theta), &(171)\end{aligned}$$

for any $f \in C(\mathbb{S}^1)$ by strong continuity of $\{P_U^t\}$ (Feller property) and continuity of $\tilde{\lambda}$. Overall we thus have that $f \in \mathcal{D}(\mathcal{L}_U)$ if and only if

$$\begin{aligned}\frac{1}{h}\mathbb{E}^\theta \left[f(U_h) - f(\theta)\right] &= \frac{f(\theta + h) - f(\theta)}{h} + \tilde{\lambda}(\theta)\left[Qf(\theta) - f(\theta)\right] + o(1) &(172)\\ &\to g(\theta) \in C(\mathbb{S}^1), &(173)\end{aligned}$$

which is clearly equivalent to $f \in C^1(\mathbb{S}^1)$.

Finally to see that $\mathrm{d}\theta/2\pi$ is invariant, having identified the domain we can easily check that for any $f \in C(\mathbb{S}^1)$ we have

$$\int \mathrm{d}\theta P_U^t f(\theta) = \int_{s=0}^t \int \mathrm{d}\theta \mathcal{L}_U P_U^s f(\theta)\mathrm{d}\theta\mathrm{d}s. \tag{174}$$

Since $f \in \mathcal{D}(\mathcal{L}_U)$ we have that $P_U^s g \in \mathcal{D}(\mathcal{L}_U)$. Since for any $g \in \mathcal{D}(\mathcal{L}_U)$ we have

$$\begin{aligned}\int \mathrm{d}\theta \mathcal{L}_U f(\theta) &= \int_{\theta=0}^{2\pi} f'(\theta)\mathrm{d}\theta + \int_{\theta=0}^{2\pi} \tilde{\lambda}(\theta)f(Q(\theta))\mathrm{d}\theta - \int_{\theta=0}^{2\pi} \tilde{\lambda}(\theta)f(\theta)\mathrm{d}\theta &(175)\\ &= f(2\pi) - f(0) + \int_{\theta=0}^{2\pi} \tilde{\lambda}(\theta)f(Q(\theta))\mathrm{d}\theta. &(176)\end{aligned}$$

□

Proposition 4. *Suppose $U^N(0)$ converges weakly to $U(0)$, then U^N converges weakly to U in $D(\mathbb{R}_+, [0,1])$.*

Proof. We will once again use Theorem 5. The generator of U_N is given by

$$\mathcal{L}_U^N f(\theta) = N\left[f(\theta + 1/N) - f(\theta)\right] s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right) + N\left[f(-\theta) - f(\theta)\right] r\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right). \tag{177}$$

We will consider the two terms separately. To this end notice that by (26), the boundedness of λ and the fact that $G \in C^1[0,1]$

$$\left|1 - s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right)\right| \leq \frac{C}{N}, \tag{178}$$

for some $C > 0$. Thus, using the mean value theorem, for each $\theta \in [0, 2\pi)$, there exists $g_N(\theta) \in [\theta, \theta + 1/N]$ such that

$$N\left[f(\theta + 1/N) - f(\theta)\right] s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right) = f'\left(g_N(\theta)\right)\left(1 + O(1/N)\right) = f'\left(\theta\right)\left(\left(1 + o(1)\right)\right), \tag{179}$$

where the errors are uniformly bounded and to obtain the second equality above we have used the fact that $|g_N(\theta) - \theta| \leq 1/N$ and that f' is uniformly continuous, being continuous on a compact set.

Overall we can see that as $N \to \infty$

$$\sup_\theta \left|N\left[f(\theta + 1/N) - f(\theta)\right] s\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right) - f'(\theta)\right| \to 0. \tag{180}$$

Next, using (26) we have that

$$r\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right) = \tilde{\lambda}(\theta)\frac{1}{N} + o(N^{-1}), \tag{181}$$

where the error is uniform in θ, whence we easily conclude that

$$N\left[f(-\theta) - f(\theta)\right] r\left(\tilde{\beta}_\theta, \tilde{\beta}_{\Sigma^N(\theta)}\right) \to \tilde{\lambda}(\theta)\left[Qf(\theta) - f(\theta)\right], \tag{182}$$

uniformly in θ. □

F.6.1 Proof of Theorem 4(b)

Now we are ready to prove the main result of this section. Notice that $Z^N(\cdot) = \phi\left(U^N(\cdot)\right)$ and $Z(\cdot) = \phi\left(U(\cdot)\right)$.

From Proposition 4 we know that the finite dimensional distributions of U_N converge to those of U. If ϕ were continuous we could conclude using the continuous mapping theorem. Since it

is not continuous at the points $\{0,1\}$, we will be using [Billingsley, 2013, Theorem 2.7]. We have to check that the law of the limiting process, that is the law of $\{U(\cdot)\}$ places zero mass on finite dimensional distributions that hit $\{0,1\}$, that is for $n \in \mathbb{N}$ and $0 < t_1 < \cdots < t_n$ we want

$$\mathbb{P}\left[U(t_i) \in \{0,1\} \text{ for some } i \in \{1,\ldots,n\}\right] = 0, \tag{183}$$

when $U(0)$ is initialized according to $\mathrm{d}\theta/2\pi$. But the above follows from the fact that $\mathbb{P}[U(t_i) \in \{0,1\}] = 0$, by stationarity when $U(0)$ is initialised uniformly on $\mathbb{S}^1$.

Relative compactness of $\{Z_N(\cdot)\}_N$ can be easily seen to follow from the compact containment condition [Ethier and Kurtz, 2009, Remark 3.7.3]. This combined with convergence of the finite dimensional distributions of Z^N to those of Z concludes the proof.

Appendix G Experiment supplements

G.1 Reproducibility.

To make our adaptive non-reversible method easy to use we implemented it as an inference engine in the open source probabilistic programming language (PPL) Blang https://github.com/UBC-Stat-ML/blangSDK. A full description of the models used in the paper are available at https://github.com/UBC-Stat-ML/blangDemos, see in particular https://github.com/UBC-Stat-ML/blangDemos/blob/master/src/main/resources/demos/models.csv for a list of command line options and data paths used for each model. All methods use the same exploration kernels, namely slice sampling with exponential doubling followed by shrinking [Neal, 2003]. Scripts documenting replication of our experiments are available at https://github.com/UBC-Stat-ML/ptbenchmark.

G.2 Multi-core implementation.

We use lightweight threads [Friesen, 2015] to parallelize both the exploration and communication phases, as shown in Algorithm 1. We use the algorithm of [Leiserson et al., 2012] as implemented in [Steele and Lea, 2013] to allow each PT chain to have its own random stream. This technique avoids any blocking across threads and hence makes the inner loop of our algorithm embarrassingly parallel in N. Moreover, the method of [Leiserson et al., 2012] combined with the fact that we fix random seeds means that the numerical value output by the algorithm is not affected by the number of threads used. Increasing the number of threads simply makes the algorithm run faster. In all experiments unless noted otherwise we use the maximum number of threads available in the host machine, by default an Intel i5 2.7 GHz (which supports 8 threads via hyper-threading) except for Section 7.4 where we use an Amazon EC2 instance of type `c4.8xlarge`, which is backed by a 2.9 GHz Intel Xeon E5-2666 v3 Processor (20 threads).

G.3 Stochastic optimization methods.

All baseline methods are implemented in Blang (https://github.com/UBC-Stat-ML/blangSDK), the same probabilistic programming language used to implement our method. The code for the baseline adaption methods are available at https://github.com/UBC-Stat-ML/blangDemos. All methods therefore run on the Java Virtual Machine, so their wall clock running times are all comparable.

Both [Atchadé et al., 2011] and [Miasojedow et al., 2013] are based on reversible PT together with two different flavours of stochastic optimization to adaptively select the annealing schedule. In [Atchadé et al., 2011], the chains are added one by one, each chain targeting a swap acceptance rate of 23% from the previous one. In [Miasojedow et al., 2013], this scheme is modified in two ways: first, all annealing parameters are optimized simultaneously, and second, a different update for performing the stochastic optimization is proposed. To optimize all chains simultaneously, the authors assume that both the number of chains and the equi-acceptance probability are specified. Since this information is not provided to the other methods, in order to perform a fair comparison, for the method we label as "Miasojedow, Moulines, Vihola" we implemented a method which adds the chain one at the time while targeting the swap acceptance rate of 23% but based on the improved stochastic optimization update of [Miasojedow et al., 2013]. Specifically, both [Atchadé et al., 2011] and [Miasojedow et al., 2013] rely on updates of the form $\rho_{n+1} = \rho_n + \gamma_n(\alpha_{n+1} - 0.23)$ where γ_n is an update schedule and ρ_n is a re-parameterization of difference in annealing parameter from the previous chain β to the one being added β'. The work of [Atchadé et al., 2011] uses the update $\beta'_n = \beta(1 + \exp(\rho_n))^{-1}$, whereas the work of [Miasojedow et al., 2013] specifies the explicit parameterization used for ρ, namely $\rho = \log(\beta'^{-1} - \beta^{-1})$, from which the update becomes $\beta'_n = \beta(1 + \beta\exp(\rho_n))^{-1}$. Moreover, while [Atchadé et al., 2011] use $\gamma_n = (n+1)^{-1}$, [Miasojedow et al., 2013] suggest to use $\gamma_n = (n+1)^{-0.6}$. We found that the latter set of choice was more stable.

G.4 Description of models and datasets

In Section 7.4 we benchmarked the methods on the following four models. First, a hierarchical model applied to a dataset of rocket launch failure/success indicator variables [McDowell, 2019]. We organized the data by types of launcher, obtaining 5,667 launches for 367 types of rockets (processed data available at https://github.com/UBC-Stat-ML/blangDemos/blob/master/data/failure_counts.csv). Each type is associated with a Beta-distributed parameter with parameters tied across rocket types, with the likelihood given by a Binomial distribution (full model specification available at https://github.com/UBC-Stat-ML/blangDemos/blob/master/src/main/java/hier/HierarchicalRockets.bl). The second model is a Spike-and-Slab variable selection model applied to the RMS Titanic Passenger Manifest dataset [Hind, 2019]. The preprocessed data is available at https://github.com/UBC-Stat-ML/blangDemos/tree/master/data/titanic. The data consist in binary classification indicators for the survival of each individual passenger as well as covariates such as age, fare paid, etc. We used a Spike-and-Slab prior with a point mass at zero and a Student-t continuous component (full model specifica-

tion available at https://github.com/UBC-Stat-ML/blangDemos/blob/master/src/main/java/glms/SpikeSlabClassification.bl). Third, we used the Ising model from Section 7.2. Finally, we also used an end-point conditioned Wright-Fisher stochastic differential equation (see, e.g., [Tataru et al., 2017]). For this last model we used synthetic data generated by the model. The specification of this last model is available at https://github.com/UBC-Stat-ML/blangSDK/blob/master/src/main/java/blang/validation/internals/fixtures/Diffusion.bl. The model-specific command line options used for all four experiments is available at https://github.com/UBC-Stat-ML/blangDemos/blob/master/src/main/resources/demos/models.csv.